



**CONTROLLING THE PRIMARY MIRROR IN
A SPACE-BASED TELESCOPE UTILIZING
AN EIGENSTRUCTURE
ASSIGNMENT TECHNIQUE**

THESIS

Kim Kyongsoo, Captain, ROK Army

AFIT/GAE/ENY/04-M10

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY**

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

AFIT/GAE/ENY/04-M10

**CONTROLLING THE PRIMARY MIRROR IN A SPACE-BASED TELESCOPE
UTILIZING AN EIGENSTRUCTURE
ASSIGNMENT TECHNIQUE**

THESIS

Presented to the Faculty

Department of Aeronautics and Astronautics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the
Degree of Master of Science in Aeronautical Engineering

Kim Kyongsoo, B.S.

Captain, ROK Army

March 2004

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GAE/ENY/04-M10

**CONTROLLING THE PRIMARY MIRROR IN A SPACE-BASED TELESCOPE
UTILIZING AN EIGENSTRUCTURE
ASSIGNMENT TECHNIQUE**

**Kim Kyongsoo, B.S.
Captain, ROK Army**

Approved:

Maj. Richard G. Cobb (Chairman)

date

Dr. Bradley S. Liebst (Member)

date

Dr. William E. Wiesel (Member)

date

Abstract

The feasibility of using a sparse array of mirrors in a satellite system is being evaluated. This study focuses on the control laws necessary for achieving the minimization of the effect of vibration which happens at the primary mirror of the satellite, by using the eigenstructure assignment technique. White Gaussian noise was assumed as the external input. AFRL has developed a 79 state model with 9 control inputs and 9 outputs, 3 sparse array mirrors, and a base. To extend the theory, a simplified model with 2 mirrors and a base which has 12 states, 4 control inputs and 8 outputs was developed. The system is not completely controllable nor observable because there are 4 rigid body modes.

The control law starts from realizing the states which are controllable or uncontrollable. Then only for the controllable states, the eigenstructure assignment technique is applied. The closed-loop eigenstructure was determined by analyzing the open-loop modes. To find the best value for the eigenstructure, Newton's line search method was applied.

The relative motions of the primary and secondary mirrors were described by wave reflection dynamics. Results show that this technique achieves a reasonable amount of control use for the objective and relates these result to those obtained from linear quadratic regulator (LQR) technique.

Acknowledgments

I would like to express my sincere appreciation to my lovely wife for her endless support and confidence. Also my thesis advisor Maj. Richard Cobb for his guidance and support throughout the course of this thesis effort as well as allowance of his lecture notes.

Kyongsoo Kim.

Table of Contents

	Page
Acknowledgments	iii
List of Figures	vii
List of Tables	viii
Abstract	ix
 I. Introduction	 1
Motivation.....	1
Background	1
Problem Statement	4
Methodology	5
Organization.....	6
 II. Developing Equations of Motion (EOM).....	 8
Physical modeling of two primary mirrors and a base.	8
Deriving the Equations of Motion	8
Deriving an Error Metric	12
Conventional Linearization.....	16
Summary	19
 III. Methodology	 20
Developing the Performance Index.....	20
Response of Linear System to White Gaussian Noise	22
Stabilizable / Detectable system	25
Method to Separate a System into Controllable and Uncontrollable System	26
Analyzing the Open-Loop Eigenstructure	28
Developing Eigenstructure Assignment	28
Eigenstructure Assignment Technique with Output Feedback	29
Eigenstructure Assignment Technique for Full-State Feedback	31
State-Space Model with Limited Measurement Data	35

	Page
Determining Desired Eigenvalues	38
Determining Desired Eigenvectors	40
Linear Quadratic Regulator (LQR)	43
Summary	43
IV. Results and Analysis	45
Open-Loop System Analysis (2-D planar model)	45
Closed-Loop System assigned 3 rd mode eigenvalue	50
Closed-Loop System assigned 4 th mode eigenvalue	52
Closed-Loop System assigned 3 rd and 4 th mode eigenvalue	53
Closed-Loop System assigned 3 rd and 4 th mode eigenvector	55
Closed-Loop System using LQR	57
Comparison between Eigenstructure Assignment Technique and LQR.....	59
Closed-loop System for the AFRL DOT system with Eigenstructure Assignment technique	60
Summary	64
V. Conclusion and Recommendations	65
Conclusion	65
Recommendations	65
Appendix A. Main Script file (Eigenstructure Assignment Technique).....	67
Appendix B. Objective Function File (Finding Eigenvalue)	72
Appendix C. Constraint Function File (Finding Eigenvalue).....	74
Appendix D. Objective Function File (Finding Eigenvector)	75
Appendix E. Constraint Function File (Finding Eigenvector)	76
Appendix F. Main Script file (LQR).....	77
Appendix G. Objective Function File (LQR)	80
Appendix H. Constraint Function File (LQR)	81

	Page
Appendix I. Main Script file (AFRL Model)	82
Appendix J. Objective Function File (AFRL Model Finding Eigenvalue)	85
Appendix K. Constraint Function File (AFRL Model Finding Eigenvalue)	87
Appendix L. Objective Function File (AFRL Model Finding Eigenvector)	88
Appendix M. Constraint Function File (AFRL Model Finding Eigenvector)	90
Bibliography.....	91

List of Figures

Figure	Page
1. Sparse array mirror.....	2
2. Ultra-lightweight primary mirror segment.....	3
3. Block diagram of Eigenstructure Assignment Process	7
4. A Planar Physical Model of a Sparse Array	9
5. Diagram Showing Coordinate Relations	10
6. Errors Resulting from Relative Vertical Movement	12
7. Errors Resulting from Mirror 1 Rotational Movement	13
8. Errors Resulting from Base Rotational Movement.....	14
9. General Closed-Loop System	20
10. Relation between Natural Frequency and Damping Frequency	39
11. Necessary Condition Set Relationship Diagram.....	42
12. Open-Loop System Singular Plot (AFRL DOT System)	46
13. Open-Loop System Singular Plot (2-D Planar Model)	47
14. Open-Loop System Mode Shape (2-D Planar Model).....	48
15. Closed-Loop System Singular Plot (3 rd mode changed)	51
16. Closed-Loop System Singular Plot (4 th mode changed).....	53
17. Closed-Loop System Singular Plot (3 rd and 4 th mode changed).....	54
18. Desired Closed-Loop Mode Shape (2-D Planar Model).....	56
19. Closed-Loop System Singular Plot (Using LQR).....	58
20. Open-Loop System Singular Plot(Reduced Model)	62
21. Closed-Loop System Singular Plot(Reduced Model)	64

List of Tables

Table	Page
1. Open-Loop System Mode	47
2. Open-Loop System Measurement RMS (2-D Planar Model).....	49
3. 3 rd Mode Eigenvalue Comparison	50
4. Closed-Loop System Measurement RMS(3 rd Mode Changed)	51
5. 4 th Mode Eigenvalue Comparison.....	52
6. Closed-Loop System Measurement RMS(4 th Mode Changed)	52
7. 3 rd and 4 th Mode Eigenvalue Comparison.....	54
8. Closed-Loop System Measurement RMS(3 rd and 4 th Mode Changed)	55
9. 3 rd and 4 th Mode Optimum Mode Shape.....	55
10. 3 rd and 4 th Mode Eigenvalue Comparison (Using LQR)	58
11. Closed-Loop System Measurement RMS(Using LQR)	58
12. Optical Performance(AFRL DOT System)	60
13. Assumed Input Noise Intensity.....	61
14. Optical Performance(Reduced Model Open-Loop System)	61
15. Optical Performance(Reduced Model Closed-Loop System).....	63

CONTROLLING THE PRIMARY MIRROR IN A SPACE-BASED TELESCOPE UTILIZING AN EIGENSTRUCTURE ASSIGNMENT TECHNIQUE

I. Introduction

Motivation

The United States derives great benefit from space-based assets, and the importance of capability of the satellite is continually increasing. In 1990, NASA developed and launched a 2.4 meter diameter primary mirror in a 600Km orbit. The effort to create large satellites and launching them into space is not an easy one. To launch large satellites we have to pay a lot of attention to drag, GNC and manufacturing issues. These requirements drive the satellites to be small. But small satellites have their own restrictions like narrow scope angles. So to achieve the advantages of large satellites, the Deployable Optical Telescope (DOT) is being considered by AFRL as the next generation of large optical satellites. To reduce weight, AFRL has selected sparse array aperture which can achieve larger diameters, instead of filled aperture. To compensate for lack of image they added high tech image processing for reconstructing the perfect image, albeit dimmer than the same-size filled aperture.

Background (Description of AFRL's Deployable Optical Telescope (DOT))

The following description was based on Development of a spare-aperture testbed for opto-mechanical control of space-deployable structures from AFRL(ref [1]).

The DOT system is a sparse aperture finite-conjugate imaging system and is shown in Figure 1 [ref (13)]. The primary mirror is a three-element, spherical, reflective sparse aperture array. The secondary mirror is a monolithic element, and is also a spherical surface of rotation. The design is similar in structure to a large space telescope with a deployable sparse-aperture primary mirror and deployable secondary. Each primary mirror segment has a spherical radius of

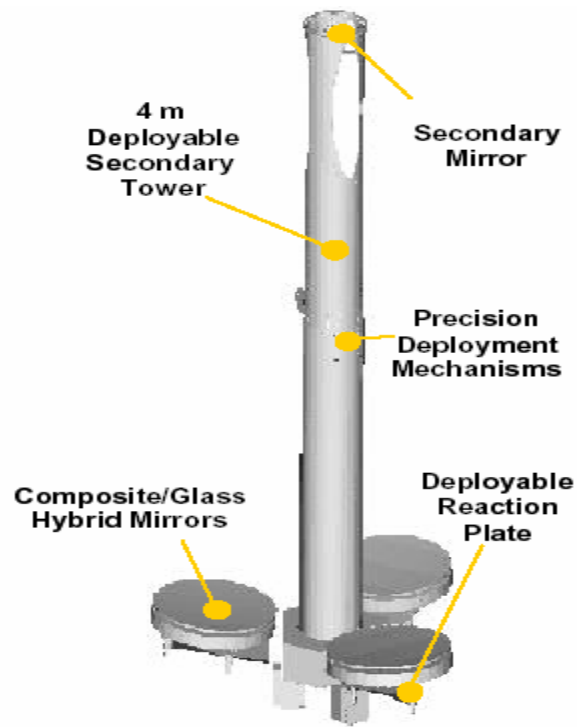


Figure 1. Sparse array mirror

curvature of 5 meters and a clear aperture of 600 mm in diameter. The vertices of these segments are located in a radially symmetric configuration, a distance of 550 mm from the primary optical axis at increments of 120 degrees. The circumscribed aperture of the assembled system is 1.7 meters, with a fill factor of 37.4 percent and an equivalent light-collecting aperture of a 1.04 meter diameter monolithic mirror.

The impetus for the finite conjugate optical design is one of cost. For the lab design with a finite object distance, a point source may be used for full aperture illumination of the primary mirror, mitigating the need for a nearly 2 meter diameter collimator in the pseudo-star illuminator system. Additionally, the purely spherical reflectors require only three degrees of alignment correction, reducing the complexity of the actuation and control sensor metrology designs. Thus, the lab design is representative of the space system, but unique.

A single lightweight primary mirror segment is shown in Figure 2. The construction consists of a thin ultra-low expansion (ULE) glass face sheet, supported by

an array of composite rods attached to a lightweight composite structure to provide stiffness. The composite portions of the structure are nickel coated to improve thermal conduction properties and reduce the distorting effects of large thermal gradients.

Each primary mirror segment is attached to a supporting deployable boom or reaction plate, by three actuator stacks. The actuator assembly consists of a low voltage piezo-ceramic stack mounted atop a micrometer drive with a DC servomotor. The piezo actuator has a range of 30 μm and a resolution of 1 nm, providing high bandwidth control capability to compensate for induced mirror jitter due to external disturbances, such as reaction wheels. These devices are driven by a 0-100v analog signal obtained from a linear amplifier. The micrometer has a 10-mm range with a 60-nm resolution, providing a low bandwidth, coarse positioning capability over a broad range, to compensate for

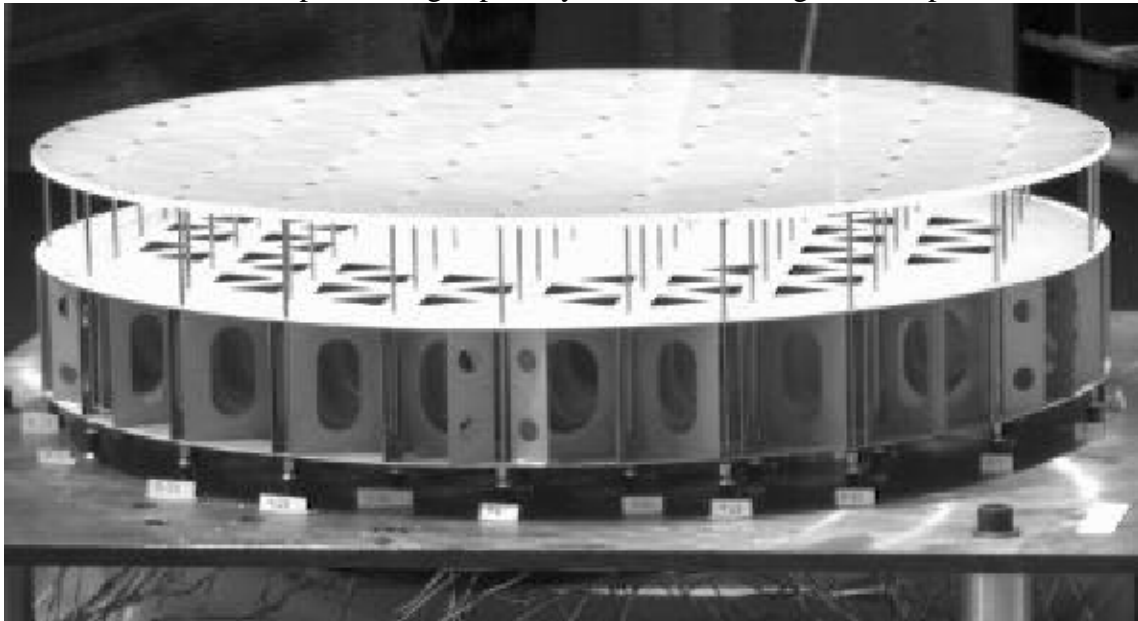


Figure 2. Ultra-lightweight primary mirror segment

deployment errors in positioning the mirror. The resolution of the coarse actuators is obtained by use of an encoder sensor built into the servomotor, which provides feedback signals to a separate PID controller. This controller accepts external commands from the control executive via an RS-232 serial interface. Each assembly is attached to the reaction

plate via a screw clamp on the micrometer. A diaphragm flexure assembly, attached to the reaction plate by three posts, protects the piezo-ceramic from shear forces induced by gravitational loading when the reaction plates are in the stowed position.

The reaction plates are constructed of lightweight composite materials. These sub-assemblies are attached to a main optics integrating structure via a stainless steel tie-rod assembly that exhibits dynamic behavior similar to a latch mechanism.

Problem Statement

Many dynamical systems are modeled using Newton's Law or Lagrange's equations and perturbation theory. The result is a second order system of linear constant coefficient differential equations. This class of systems can be mathematically described by the equations of motions

$$M\ddot{x} + C\dot{x} + Kx = Du \quad (1)$$

where $x \in \mathfrak{R}^n$ and $u \in \mathfrak{R}^m$ are the state and control (actuator) vectors respectively, M is the $n \times n$ positive definite symmetric mass matrix, C is the $n \times n$ positive semi-definite symmetric structural damping matrix, K is the $n \times n$ positive semi-definite symmetric stiffness matrix, D is the $n \times m$ control influence matrix, and $\dot{x} = \frac{dx}{dt}$.

By using output feedback, the control and measurement equation can be written as

$$y = C_m x \quad (2)$$

$$\dot{y} = C_m \dot{x} \quad (3)$$

$$u = -G_p y - G_v \dot{y} \quad (4)$$

where $y \in \Re^m$ is the output(sensor) vector, and C_m is the $m \times n$ output measurement matrix, and G_p and G_v are the $n \times m$ feedback gains. Equation (2), (3) and (4) can be substituted into Equation (1) and everything can be taken to the left hand side:

$$M\ddot{x} + (C + D G_v C_m)\dot{x} + (K + D G_p C_m)x = 0 \quad (5)$$

The problem considered herein is how to select the control matrices $[G_v, G_p]$, such that it minimizes a cost function. The system considered contains both rigid body and flexible modes. The control matrices are selected based on a placing the eigenvalues and eigenvectors (hereafter referred to as the eigenstructure) such that the achieved eigenstructure minimizes the cost function.

Methodology

The research for this thesis includes developing mathematical models representative of a sparse telescope array and designing a technique and computer algorithm for designing a feedback control system. The desired eigenvalues and eigenvector (eigenstructure) were found using Newton's line search method by analyzing the open-loop system and the eigenstructure assignment technique is used to find the proper control gain matrix. To place realistic limits on the control usage, the maximum element value of the control gain matrix was constrained.

The results of the eigenstructure assignment technique were compared with results obtained using a linear quadratic regulator (LQR) approach.

Organization

This thesis is organized around designing a control system for a dynamic system with rigid body and flexible modes. Figure 3 illustrates the overall process

This thesis begins with developing the equations of motion (EOM) for the simplified satellite model at Chapter 2. Chapter 3 establishes the techniques for achieving the proper control gain for the eigenstructure assignment technique. Chapter 3 also addresses the technique for systems that do not have enough measurement information to place all the desired eigenvalues. These results are then compared to LQR results. Then Chapter 4 provides result applied to the theoretical model. Lastly, the technique is applied to AFRL's DOT experimental data.

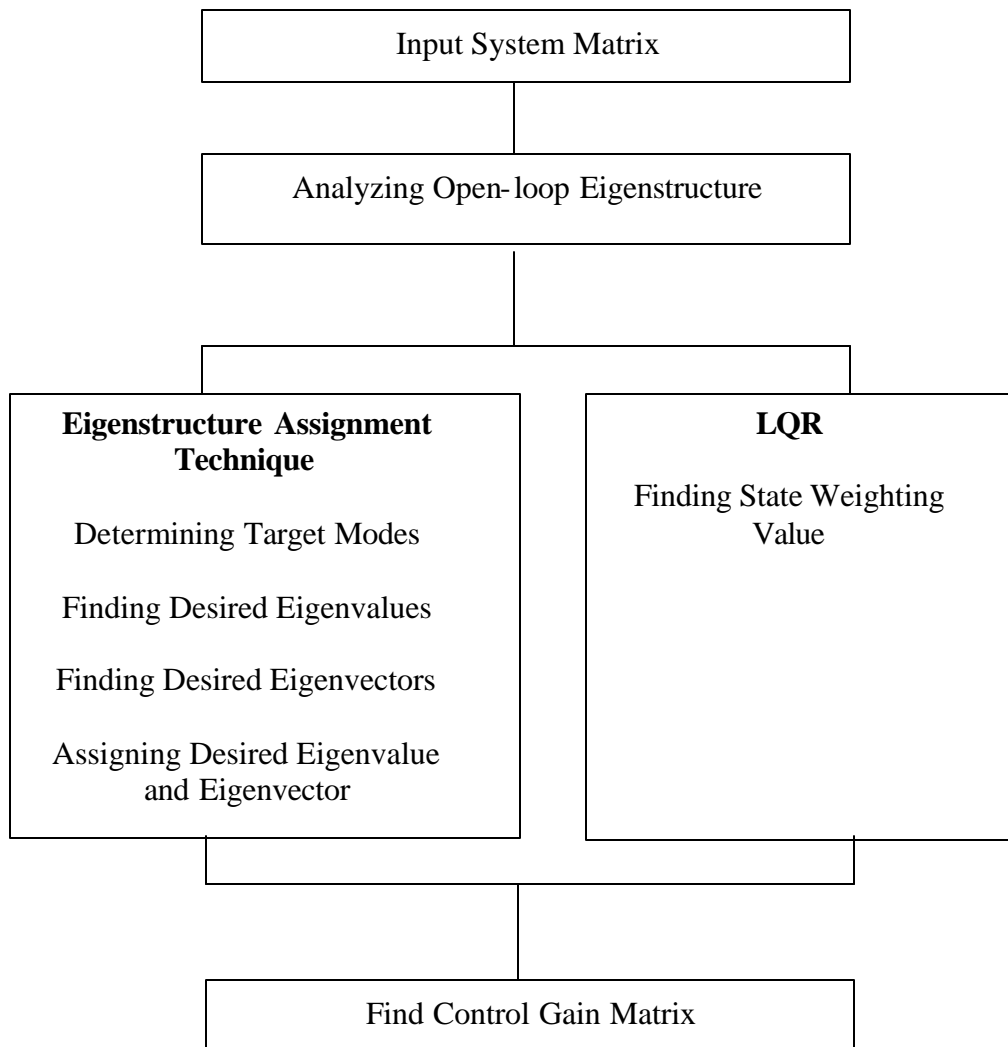


Figure 3. Block diagram of Eigenstructure Assignment Process

II. Developing Equations of Motion

Physical modeling of two primary mirrors and a base

The DOT system is three dimensional. To demonstrate the technique and aid in analysis, a simplified planar model representative of a system with two primary mirrors and a base was used. For this model, as in the actual structure, all the elements are subject to small perturbations, and thus the small angle and small disturbance approximations were used.

Each mirror's movement has both horizontal and vertical movement. Because of the small angle approximation, horizontal movement will be negligible with respect to vertical movement, thus only vertical directional movement and rotation was considered. As a result, each mirror has 2 degrees-of-freedom, namely the vertical movement and rotation. The planar model is shown in Figure 4. The full system has a total of 6 degrees-of-freedom. Although there are six DOFs, only internal force actuators are available, thus not all the DOFs can be controlled. For this model, as with DOT, the rigid body modes can not be controlled.

For system control, there are two actuators for each mirror and each actuator has its own spring damper and electronic forcing device such as the piezo actuators used on DOT.

Deriving the Equations of Motion.

Mirror "1" and "2" have mass m_1 and m_2 and moment of inertia I_1 and I_2 respectively. Next, define the rotation angle of mirror "1" and "2" as θ_1 and θ_2 with counter-clockwise direction for positive rotations. In this expression A_1 , A_2 , A_3 and A_4 stand for the actuators that have a spring, a viscous damper, and an electronic forcing

device in parallel as shown in Figure 4. This is typical for space applications, where the flexible modes are controlled, independent from the rigid body (orbital motion).

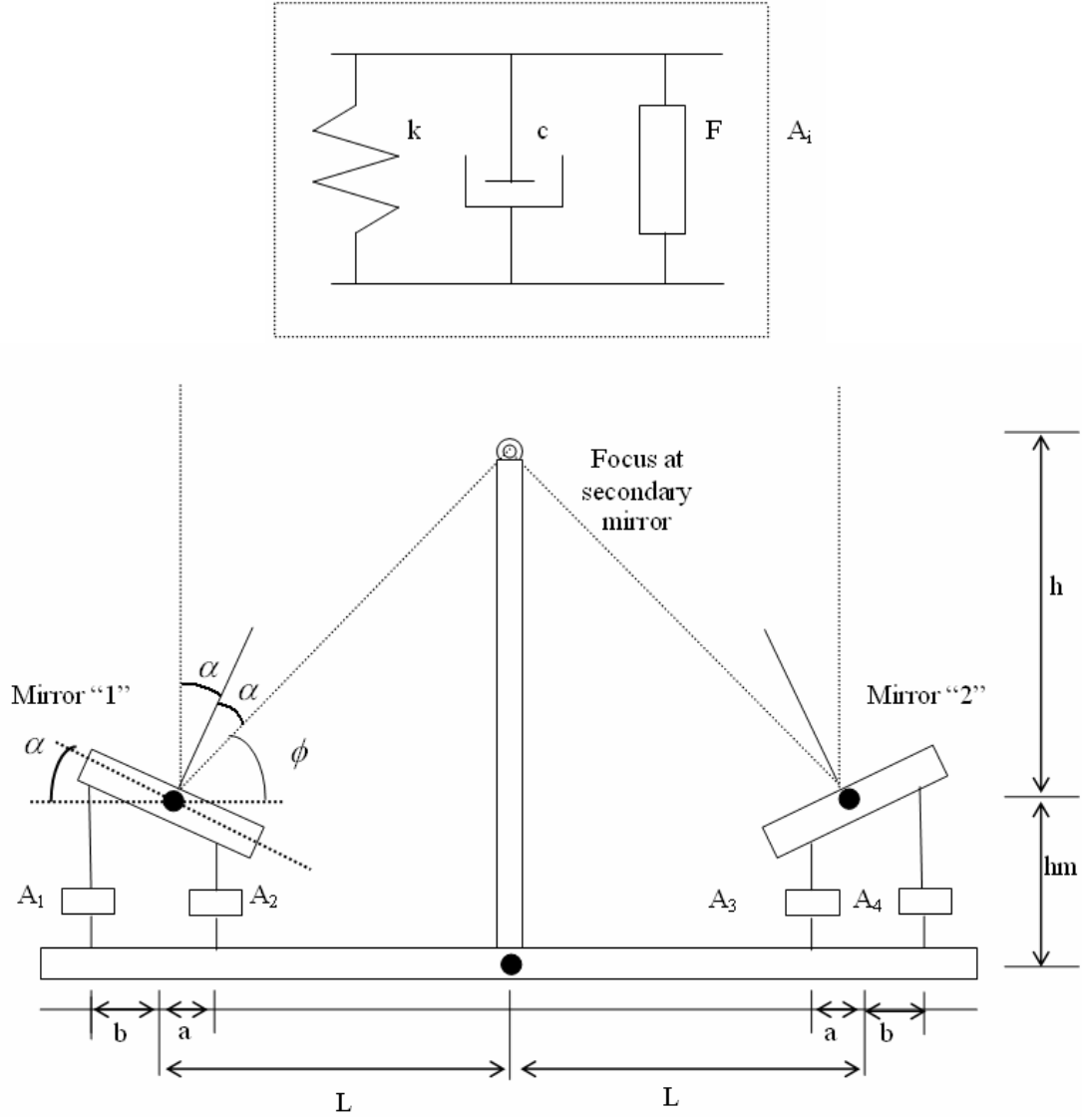


Figure 4. A Planar Physical Model of a Sparse Array

The equations of motion at each actuator is:

$$A_i = k_i D_i + c_i \dot{D}_i + F_i \quad (6)$$

where $i = 1, 2, \dots, 4$, D_i represents the relative distance in the actuator and $\dot{D}_i = \frac{dD_i}{dt}$.

For this analysis, the elements representing the base, mirrors and secondary mirror were assumed as rigid elements.

From a static point of view, for the mirror to focus on the focal point, the angle should be :

$$f = \arctan\left(\frac{h}{L}\right) \quad (7)$$

$$a = \frac{p}{4} - \frac{f}{2} \quad (8)$$

where h and L are defined as in Figure 4.

To derive the EOMs, first consider the free body diagram for mirror “1”.

If the mirror “1” and the base were disturbed in the positive direction then the transversal movement will be as shown in Figure 5.

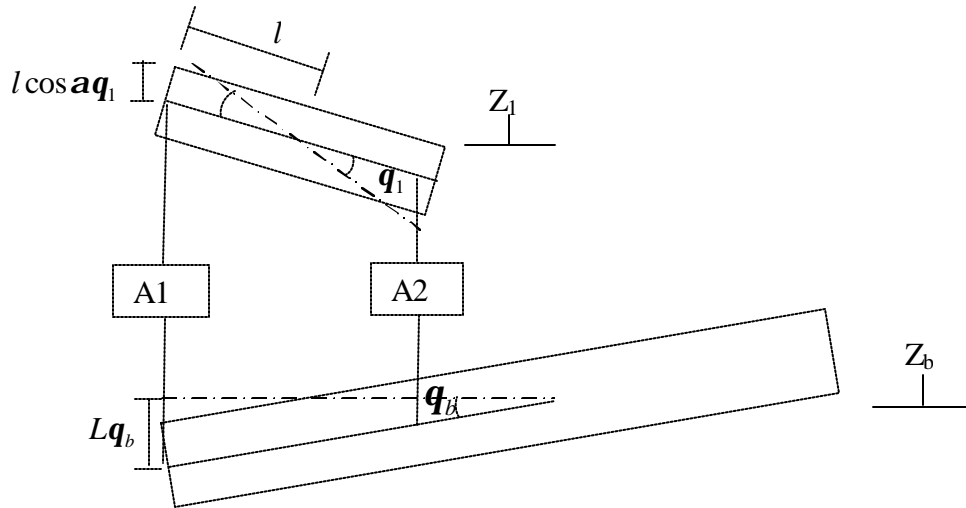


Figure 5. Diagram Showing Coordinate Relations

Thus the distance difference (D_i) for the actuator 1 and 2 will be:

$$\begin{aligned} D_1 &= z_1 - z_b - l\mathbf{q}_1 \cos \mathbf{a} + (L + l \cos \mathbf{a})\mathbf{q}_b \\ D_2 &= z_1 - z_b + l\mathbf{q}_1 \cos \mathbf{a} + (L - l \cos \mathbf{a})\mathbf{q}_b \end{aligned} \quad (9)$$

where l and \mathbf{q}_i are defined in Figure 5.

With this distance difference and using Equation (6) each actuator will have a force of :

$$\begin{aligned} A_1 &= k_1 D_1 + c_1 \dot{D}_1 + F_1 \\ A_2 &= k_2 \dot{D}_2 + c_2 \dot{D}_2 + F_2 \end{aligned} \quad (10)$$

Notice that the positive distance difference will produce a compressive force.

The resulting Equations of Motion for mirror 1 is (from direct application of Newton's Law):

$$\begin{aligned} m_1 \ddot{z}_1 &= -A_1 - A_2 \\ I_1 \ddot{\mathbf{q}}_1 &= l \cos \mathbf{a} (A_1 - A_2) \end{aligned} \quad (11)$$

Likewise for mirror 2, the distance difference for actuator 3 and 4 will be:

$$\begin{aligned} D_3 &= z_2 - z_b - l\mathbf{q}_2 \cos \mathbf{a} - (L - l \cos \mathbf{a})\mathbf{q}_b \\ D_4 &= z_2 - z_b + l\mathbf{q}_2 \cos \mathbf{a} - (L + l \cos \mathbf{a})\mathbf{q}_b \end{aligned} \quad (12)$$

and each actuator will have a force of:

$$\begin{aligned} A_3 &= k_3 D_3 + c_3 \dot{D}_3 + F_3 \\ A_4 &= k_4 D_4 + c_4 \dot{D}_4 + F_4 \end{aligned} \quad (13)$$

In a similar fashion, the Equations of Motion for mirror 2 and the base will be:

$$\begin{aligned}
m_2 \ddot{z}_2 &= -A_3 - A_4 \\
m_b \ddot{z}_b &= A_1 + A_2 + A_3 + A_4 \\
I_2 \ddot{\mathbf{q}}_2 &= l \cos \mathbf{a} (A_3 - A_4) \\
I_b \ddot{\mathbf{q}}_b &= [L - l \cos \mathbf{a}] (A_3 - A_2) + [L + l \cos \mathbf{a}] (A_4 - A_1)
\end{aligned} \tag{14}$$

Deriving an Error Metric

With this set of dynamic equations, we define our output error (which consist of the two tilt angles and the two wavelength errors) as follows.

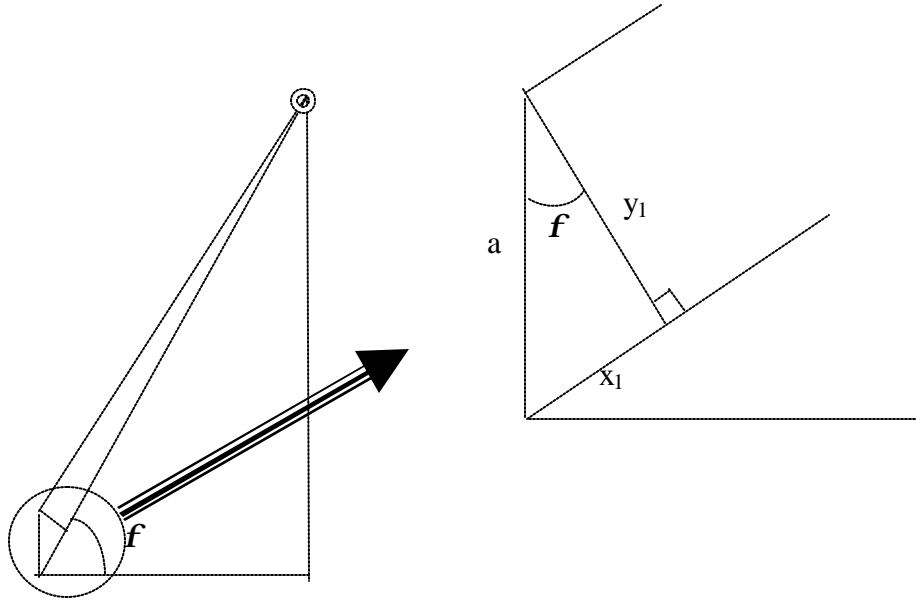


Figure 6. Errors Resulting from Relative Vertical Movement.

The tilt angle errors and wavelength errors are developed separately. The total error is then the sum of all the errors. From Figure 6 the distance x_1 and y_1 resulting from vertical movement is:

$$x_1 = a \sin \mathbf{f} \tag{15}$$

$$y_1 = a \cos \mathbf{f} \tag{16}$$

where $a = z_1 - z_b$

Then the wavelength error (ER_1) and tilt angle error (T_1) for mirror 1 is:

$$ER_1 = -x_1 = -a \sin \mathbf{f} \quad (17)$$

$$T_1 = \frac{y_1}{\sqrt{L^2 + h^2}} = \frac{a \cos \mathbf{f}}{\sqrt{L^2 + h^2}} \quad (18)$$

Likewise the wavelength error (ER_2) and tilt angle error (T_2) for mirror 2 is:

$$ER_2 = -b \sin \mathbf{f} \quad (19)$$

$$T_2 = -\frac{b \cos \mathbf{f}}{\sqrt{L^2 + h^2}} \quad (20)$$

where $b = z_2 - z_b$

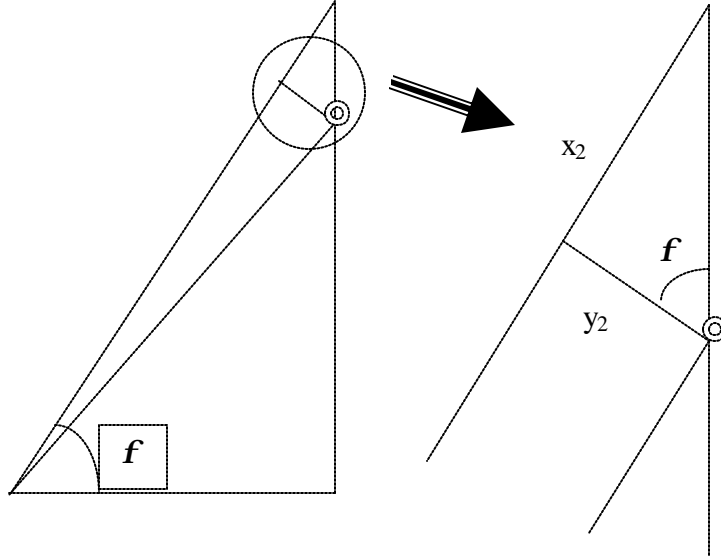


Figure 7. Errors Resulting from Mirror 1 Rotational Movement

From Figure 7 the distance x_2 and y_2 made by rotational movement of mirror 1 is:

$$y_2 = \sqrt{L^2 + h^2} \mathbf{q}_1 \quad (21)$$

$$x_2 = y_2 \tan \mathbf{f} \quad (22)$$

Then the wavelength error (ER_1) and tilt angle error (T_1) for mirror 1 is:

$$ER_1 = x_2 = \tan f \sqrt{L^2 + h^2} q_1 \quad (23)$$

$$T_1 = q_1 \quad (24)$$

Likewise the wavelength error (ER_2) and tilt angle error (T_2) for mirror 2 is:

$$ER_2 = -\tan f \sqrt{L^2 + h^2} q_2 \quad (25)$$

$$T_2 = q_2 \quad (26)$$

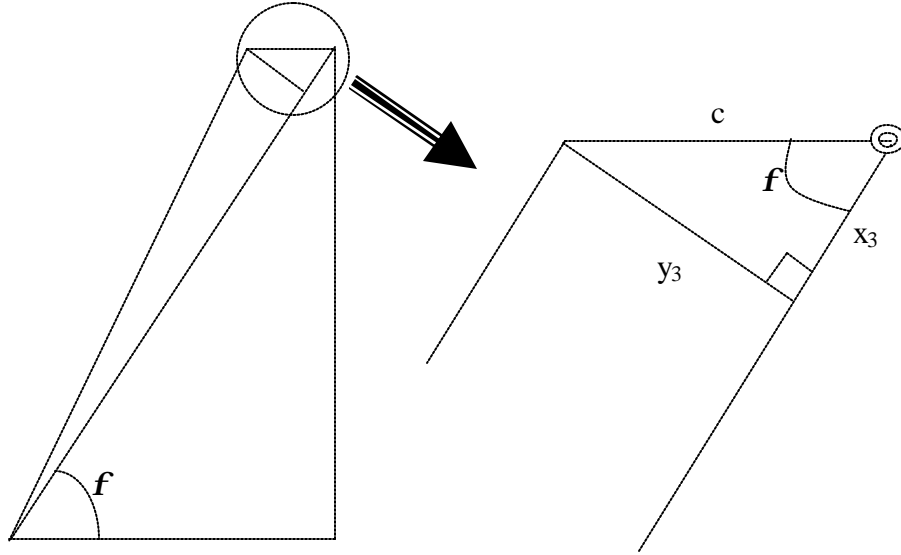


Figure 8. Error Resulting from Base Rotational Movement

From Figure 8, the distance x_3 and y_3 made by horizontal movement is:

$$x_3 = c \cos f \quad (27)$$

$$y_3 = c \sin f \quad (28)$$

where $c = (h + h_m) q_b$

The wavelength error (ER_1) and tilt angle error (T_1) for mirror 1 is:

$$ER_1 = -x_3 = -c \cos \mathbf{f} \quad (29)$$

$$T_1 = -\frac{y_3}{\sqrt{L^2 + h^2}} = -\frac{c \sin \mathbf{f}}{\sqrt{L^2 + h^2}} \quad (30)$$

Likewise the wavelength error (ER2) and tilt angle error (T2) for mirror 2 is:

$$ER_2 = x_3 = c \cos \mathbf{f} \quad (31)$$

$$T_2 = -\frac{y_3}{\sqrt{L^2 + h^2}} = -\frac{c \sin \mathbf{f}}{\sqrt{L^2 + h^2}} \quad (32)$$

The total wavelength error and total tilt angle error made by the vertical and horizontal movement will be the sum of all the above errors. In addition to this, the tilt angle error is changed by rotational movements of mirror 1 and 2.

$$ER_1 = -(z_1 - z_b) \sin \mathbf{f} + \tan \mathbf{f} \sqrt{L^2 + h^2} \mathbf{q}_1 - (h + h_m) \mathbf{q}_b \cos \mathbf{f} \quad (33)$$

$$T_1 = \frac{(z_1 - z_b) \cos \mathbf{f}}{\sqrt{L^2 + h^2}} - \frac{(h + h_m) \mathbf{q}_b \sin \mathbf{f}}{\sqrt{L^2 + h^2}} + \mathbf{q}_1 \quad (34)$$

$$ER_2 = -(z_2 - z_b) \sin \mathbf{f} - \tan \mathbf{f} \sqrt{L^2 + h^2} \mathbf{q}_2 + (h + h_m) \mathbf{q}_b \cos \mathbf{f} \quad (35)$$

$$T_2 = -\frac{(z_2 - z_b) \cos \mathbf{f}}{\sqrt{L^2 + h^2}} - \frac{(h + h_m) \mathbf{q}_b \sin \mathbf{f}}{\sqrt{L^2 + h^2}} + \mathbf{q}_2 \quad (36)$$

This represents the coupling of errors between wavelength error and tilt angle error.

Conventional Linearization

For this analysis, we require a linearized model described in state-space form as:

$$\begin{aligned}\dot{X} &= AX + BU \\ Y &= CX + DU\end{aligned}\tag{37}$$

where the state vector X is defined as follows:

$$X = [z_1 \ z_2 \ z_b \ \mathbf{q}_1 \ \mathbf{q}_2 \ \mathbf{q}_3 \ \dot{z}_1 \ \dot{z}_2 \ \dot{z}_b \ \dot{\mathbf{q}}_1 \ \dot{\mathbf{q}}_2 \ \dot{\mathbf{q}}_3]^T = [x \ \dot{x}]^T$$

From the Equations (9) and (12), distance vector can be described as:

$$D = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & -l \cos \mathbf{a} & 0 & L + l \cos \mathbf{a} \\ 1 & 0 & -1 & l \cos \mathbf{a} & 0 & L - l \cos \mathbf{a} \\ 0 & 1 & -1 & 0 & -l \cos \mathbf{a} & -L + l \cos \mathbf{a} \\ 0 & 1 & -1 & 0 & l \cos \mathbf{a} & -L - l \cos \mathbf{a} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_b \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_b \end{bmatrix} = D_m x \tag{38}$$

In a similar fashion \dot{D} will be similar type of D .

$$\dot{D} = \begin{bmatrix} \dot{D}_1 \\ \dot{D}_2 \\ \dot{D}_3 \\ \dot{D}_4 \end{bmatrix} = D_m \dot{x} \tag{39}$$

The actuator vector will be (using Equation (10) and (13)):

$$\begin{aligned}
A_c &= \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{bmatrix} = \begin{bmatrix} k_1 & 0 & 0 & 0 & c_1 & 0 & 0 & 0 \\ 0 & k_2 & 0 & 0 & 0 & c_2 & 0 & 0 \\ 0 & 0 & k_3 & 0 & 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & k_4 & 0 & 0 & 0 & c_4 \end{bmatrix} \begin{bmatrix} D \\ \dot{D} \end{bmatrix} + \begin{bmatrix} F1 \\ F2 \\ F3 \\ F4 \end{bmatrix} \\
&= A_{cm} \begin{bmatrix} D \\ \dot{D} \end{bmatrix} + \begin{bmatrix} F1 \\ F2 \\ F3 \\ F4 \end{bmatrix}
\end{aligned} \tag{40}$$

Combining the results we can build Equations of Motion from Equation (11) and (14) in matrix form as:

$$\begin{aligned}
\begin{bmatrix} \ddot{z}_1 \\ \ddot{z}_2 \\ \ddot{z}_b \\ \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_b \end{bmatrix} &= \begin{bmatrix} -1/m_1 & -1/m_1 & 0 & 0 \\ 0 & 0 & -1/m_2 & -1/m_2 \\ 1/m_b & 1/m_b & 1/m_b & 1/m_b \\ l \cos \mathbf{a} / I_1 & -l \cos \mathbf{a} / I_1 & 0 & 0 \\ 0 & 0 & l \cos \mathbf{a} / I_2 & -l \cos \mathbf{a} / I_2 \\ -(L + l \cos \mathbf{a}) / I_b & -(L - l \cos \mathbf{a}) / I_b & (L - l \cos \mathbf{a}) / I_b & (L + l \cos \mathbf{a}) / I_b \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{bmatrix} \\
&= DD_m \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{bmatrix}
\end{aligned} \tag{41}$$

The output equation is defined in terms of Equations (33), (34), (35) and (36) as:

$$Y_1 = [T_1 \quad T_2 \quad ER_1 \quad ER_2]^T$$

where T_1 and T_2 represent the tilt angle error for the mirror 1 and 2 and ER_1 and ER_2 represent wavelength error for the mirror 1 and 2.

In matrix form this is represented as:

$$Y = C_m \begin{bmatrix} z_1 \\ z_2 \\ z_b \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_b \end{bmatrix} = C_m x \quad (42)$$

where

$$C_m = \begin{bmatrix} \frac{\cos \mathbf{f}}{\sqrt{L^2 + h^2}} & 0 & -\frac{\cos \mathbf{f}}{\sqrt{L^2 + h^2}} & 1 & 0 & -\frac{(h+h_m)\sin \mathbf{f}}{\sqrt{L^2 + h^2}} \\ 0 & -\frac{\cos \mathbf{f}}{\sqrt{L^2 + h^2}} & \frac{\cos \mathbf{f}}{\sqrt{L^2 + h^2}} & 0 & 1 & -\frac{(h+h_m)\sin \mathbf{f}}{\sqrt{L^2 + h^2}} \\ -\sin \mathbf{f} & 0 & \sin \mathbf{f} & \tan \mathbf{f} \sqrt{L^2 + h^2} & 0 & -(h+h_m)\cos \mathbf{f} \\ 0 & -\sin \mathbf{f} & \sin \mathbf{f} & 0 & -\tan \mathbf{f} \sqrt{L^2 + h^2} & (h+h_m)\cos \mathbf{f} \end{bmatrix} \quad (43)$$

Thus the output equation is given as:

$$Y = [C_m \quad 0]X \quad (44)$$

$$\text{where } X = [z_1 \quad z_2 \quad z_b \quad \mathbf{q}_1 \quad \mathbf{q}_2 \quad \mathbf{q}_b \quad \dot{z}_1 \quad \dot{z}_2 \quad \dot{z}_b \quad \dot{\mathbf{q}}_1 \quad \dot{\mathbf{q}}_2 \quad \dot{\mathbf{q}}_b]^T$$

Now in the state-space form, the A, B, C and D matrix will be:

$$A = \begin{bmatrix} \text{zero}_{6 \times 6} & I_{6 \times 6} \\ DD_m \cdot A_{cm} \cdot \begin{bmatrix} D_m & \text{zero}_{4 \times 6} \\ \text{zero}_{4 \times 6} & D_m \end{bmatrix} \end{bmatrix}_{12 \times 12} \quad (45)$$

where $I(6 \times 6)$ stands for identity matrix.

$$B = \begin{bmatrix} zero_{6 \times 4} \\ DD_m \end{bmatrix} \quad (46)$$

$$C = [C_m \quad 0] \quad (47)$$

and

$$D = zero_{4 \times 4} \quad (48)$$

For a given set of physical parameters of the mirror, base and actuators, generation of the corresponding state-space model matrices A, B, C and D have been automated and the MATLAB code is contained in Appendix A.

Summary

In this chapter a simplified two dimensional DOT system and the corresponding EOMSs were developed. The state-space model has less measurement outputs and control inputs than the number of states. Chapter 3 establishes the techniques for achieving the proper gains for the eigenstructure assignment technique. The eigenstructure assignment technique and the constraints will be discussed on a case by case basis.

III. Methodology

The previous chapter developed the simplified planar model. This chapter develops the methodology for control. It begins with a development of a performance index for the system. Eigenstructure assignment technique will be developed in the case of the full-state feedback and the output feedback. Then the LQR methods will be reviewed to compare the result.

Developing the performance index

Control system performance is typically measured with respect to a performance metric. For the problem herein, a suitable metric must be established to measure the closed-loop performance. For a linear system as given below:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned}\tag{49}$$

Under proportional output feedback we get the closed-loop system shown in Figure 9.

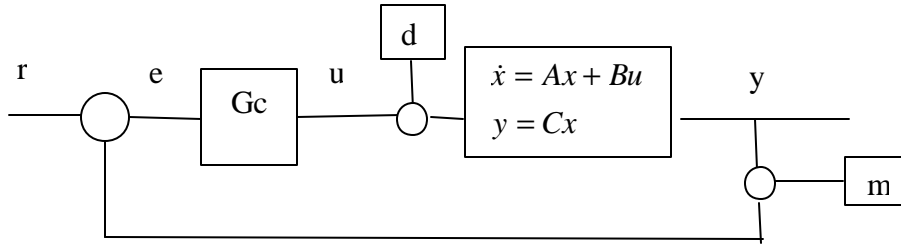


Figure 9. General Closed-loop System

For the sparse array, the objective is to minimize the effect of disturbances (d) when $r = 0$. This is referred to as the regulator problem, i.e. ($r = 0$ above) the objective of

the regulator is to drive $e \rightarrow 0$ and thus $y \rightarrow 0$ (or $x \rightarrow 0$ if full-state feedback $C = I$ is available) . The effect of measurement error (m) will be not considered in the research.

One measure of how successful the controller is

$$J_1 = \int_0^{t_f} x^T Q x dt \quad (50)$$

where Q is a positive semidefinite scalar weighting matrix such as

$$Q = \text{diag}(q_1, q_2, q_3, \dots, q_n) \quad (51)$$

For a fixed cost J_1 the bigger we make q_i the smaller x_i must be, and thus the speed of response will be improved by increasing a particular q .

In general we can always speed up the all response by increasing the gain but it will cost more control energy to achieve. So we probably need to add a penalty for control use. Thus consider

$$J_2 = \int_0^{t_f} (x^T Q x + u^T R u) dt \quad (52)$$

where R is positive definite. A natural choice would be

$$R = \text{diag}(r_1, r_2, r_3, \dots, r_m) \quad (53)$$

and if we pick r_i large, then u_i would be small for a given J_2 . Thus we can see that by varying our choices in Q and R , we can trade off speed of response for control use.

If there is specified external input (d in Figure (9)), the objective of control system is minimizing the effect of the disturbance (noise). For the case of white Gaussian

Noise, every output variance value from the noise can be calculated through a Lyapunov Equation. If output variance is defined such as $Vy = [v_{y1}, v_{y2}, \dots, v_{yr}]^T$ then the objective function can be the sum of each variance. This thesis used the sum of all output variances for the objective function.

Response of Linear system to white noise

The following is based on reference (11).

Suppose that $x(t)$ is the solution of

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bw(t) \\ x(t_0) &= x_0\end{aligned}\tag{54}$$

Where $w(t)$ is white noise with intensity $V(t)$ and x_0 is a stochastic variable independent of $w(t)$, and mean of $x(t_0)$ is equal to m_0 and the variance matrix Q_0 is

$$Q_0 = E\{(x_0 - m_0)(x_0 - m_0)^T\}\tag{55}$$

Then $x(t)$ has mean

$$m_x(t) = e^{A(t-t_0)}m_0\tag{56}$$

and covariance

$$R_x(t_1, t_2) = e^{A(t_1-t_0)}Q_0e^{A^T(t_2-t_0)} + \int_{t_0}^{\min(t_1, t_2)} e^{A(t_1-t)}BV(t)B^Te^{A^T(t_2-t)}dt\tag{57}$$

The variance matrix $Q(t)=R_x(t,t)$ satisfies the differential equation (Lyapunov)

$$\dot{Q}(t) = A Q(t) + Q(t) A^T + B V(t) B^T \quad (58)$$

where $Q(t_0)=Q_0$

Now consider the case where $w(t)$ is stationary i.e. constant intensity. If A is asymptotically stable the variance matrix will be as follows as t increases:

$$\lim_{t \rightarrow \infty} Q(t) = \lim_{t_0 \rightarrow -\infty} Q(t) = \bar{Q} = \int_0^\infty e^{A t} B V B^T e^{A^T t} dt \quad (59)$$

Further, since $Q(t)$ is the solution to the differential equation, its limit \bar{Q} must also satisfy the differential equation at steady state. Specifically

$$A \bar{Q} + \bar{Q} A^T + B V B^T = 0 \quad (60)$$

The Lyapunov equation has a unique solution for an asymptotically stable system, which means that the all of the eigenvalues of matrix A cannot be zeros [ref (12)].

For the problem considered herein, this means if $w(t)$ is a stationary, zero-mean, white Gaussian noise, and is the only input to the system

$$\dot{x}(t) = A x(t) + B w(t) \quad (61)$$

then the process $x(t)$ will also be zero-mean Gaussian noise with variance $Q(t)$, given by

$$Q(t) = e^{A(t-t_0)} Q_0 e^{A^T(t-t_0)} + \int_{t_0}^t e^{A(t-\tau)} B V(\tau) B^T e^{A^T(t-\tau)} d\tau \quad (62)$$

For the steady-state case, and stationary $w(t)$

$$\bar{Q} = E\{x(t)x^T(t)\} \quad (63)$$

and is the solution to

$$A\bar{Q} + \bar{Q}A^T + BVB^T = 0 \quad (64)$$

Further, if $y(t) = Cx(t)$ then $E\{y(t)\} = 0$ and

$$\begin{aligned} E\{y(t)y^T(t)\} &= E\{Cx(t)x^T(t)C^T\} \\ &= CE\{x(t)x^T(t)\}C^T \\ &= C\bar{Q}C^T \end{aligned} \quad (65)$$

So the root mean square response of $y(t)$ is the square root of the diagonal of the expression $C\bar{Q}C^T$. In this thesis, the objective function used is diagonal sum of $C\bar{Q}C^T$.

$$J = \sum \text{diag}(C\bar{Q}C^T) \quad (66)$$

where \bar{Q} is the intensity of the disturbance, assumed to be zero-mean White Gaussian Noise.

Stabilizable/ Detectable System

The goal of control is to stabilize the system with minimum input control. To ensure a solution exists, we must first verify that the system is both stabilizable and detectable. For a system, if there exist uncontrollable modes and the uncontrollable modes are unstable then there is no way to stabilize the system. Likewise, if there exist unobservable modes and the unobservable modes were totally undetectable, then there is no way to detect the states. Thus before a control system can be designed, we need to verify whether the system is stabilizable and detectable or not.

If the system is stabilizable and detectable, then it is reasonable to find a controller to minimize the cost function in Equation (66).

The definition of stabilizable is “if all the unstable modes are controllable then the system is said to be stabilizable”. Clearly a controllable system is sufficient to be stabilizable but the reverse is not.

The definition of detectable is “if all the unstable modes are observable, then the system is said to be detectable”. Clearly an observable system is sufficient to be detectable but the reverse is not [ref (11)].

Thus to stabilize the system, all the unstable modes must be observable in the performance index. Intuitively, if the unstable modes are not included in the performance index then the optimization will not consider them in the optimal solution, even if they are controllable.

As developed in Chapter 2, the planar model has both uncontrollable and unobservable modes. These will need to be removed prior to control design.

Method to separate a system into the controllable and uncontrollable matrix

In general, a system can have both controllable and uncontrollable modes. But to control the system we need to separate a system into the controllable and uncontrollable form. If the uncontrollable modes are stable then the system can be stabilized.

For a given system:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned}\tag{67}$$

Define the controllability matrix as [ref(2)]

$$M_c = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B]\tag{68}$$

Using singular value decomposition M_c can be decomposed as follows :

$$T\Sigma V = M_c\tag{69}$$

where the T represents an orthogonal matrix and the columns of T correspond to the singular values of M_c in Σ . Thus if M_c has m nonzero singular values and the remaining m+1....n form a basis for the rest of \mathbb{R}^n .

Then apply transformation T such as

$$\bar{A} = T^{-1}AT, \quad \bar{B} = T^{-1}B, \quad \bar{C} = CT.\tag{70}$$

so that:

$$\bar{A} = \begin{bmatrix} Acc & Acu \\ 0 & Auu \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} Bcc \\ 0 \end{bmatrix} \text{ and } \bar{C} = [Cc \quad Cu].\tag{71}$$

With this new state-space relation the output feedback system with $u = -Ky$ is

$$\dot{\bar{x}} = \left(\begin{bmatrix} Acc & Acu \\ 0 & Auu \end{bmatrix} - \begin{bmatrix} Bcc \\ 0 \end{bmatrix} K \begin{bmatrix} Cc & Cu \end{bmatrix} \right) \bar{x} \quad (72)$$

i.e.

$$\dot{\bar{x}} = \begin{bmatrix} Acc - BccKCc & Acu - BccKCc \\ 0 & Auu \end{bmatrix} \bar{x} \quad (73)$$

From the block diagonal form, it is clear that the closed-loop eigenvalue is eigenvalue of $Acc - BccKCc$ and eigenvalue of Auu .

It is also clear from Equation (72) that Auu cannot be affected by any input. So Auu results in completely uncontrollable modes. Thus if any uncontrollable modes are unstable, i.e. if real part of any eigenvalue of Auu is greater than zero, then the system is not stabilizable.

Likewise Acc can be called the completely controllable matrix.

So the system with (Acc, Bcc, Ccc) is a completely controllable system.

The state x can be expressed such as $x = T\bar{x}$, then \bar{x} can be decomposed such as $\bar{x} = \begin{bmatrix} x_c \\ x_u \end{bmatrix}$. Where x_c is controllable states and x_u is uncontrollable states.

Notice that the transfer functions $Ccc(sI - Acc)^{-1}Bcc = C(sI - A)^{-1}B$ are equivalent for the controllable subsystem and the original system. So we can use Acc , Bcc and Ccc matrix instead of A , B and C . For the model developed in Chapter 2, this technique must be applied to remove the uncontrollable and unobservable (rigid body modes) before proceeding to design the controller.

Analyzing the Open-loop Eigenstructure

In a given system

$$\begin{aligned} \begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix} &= A \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + Bu = \begin{bmatrix} 0 & I \\ -Ks & -Cd \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + \begin{bmatrix} 0 \\ Bs \end{bmatrix} u \\ y &= C \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + Du = [C_m \quad 0] \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \end{aligned} \quad (74)$$

where the system is proper and Ks represents normalized stiffness matrix, Cd represents normalized damping matrix, Bs represents input direction matrix and Cm represents the states output.

For a given system, if all the springs and dampers are collocated, then the possible open-loop mode shapes can be found from the eigenvector set of Ks or Cd . If the open-loop eigenstructure doesn't contain rigid body modes, then all the eigenvalues of Ks are non-zeros. The eigenvectors related with the zero eigenvalues stand for rigid body mode shapes and the others with non-zero eigenvalues are the mode shapes that can be controlled. By using the controllable mode shapes as the basis, any closed-loop mode shape can be obtained, constrained only by the max allowable control use. That means the basis of open-loop mode shapes are the same as the basis of closed-loop mode shapes.

Developing eigenstructure assignment [ref (11)]

Consider a system that has n states, m control inputs, and r measurement outputs. For a static output feedback system

$$u = -Ky \quad (75)$$

then the closed-loop regulator is given by :

$$\dot{x} = (A - BKC)x \quad (76)$$

and the eigenstructure is determined by $(A - BKC)$

As described before, the system can have both controllable and uncontrollable modes at a same time. Eigenstructure assignment can be used for the controllable modes.

$$\begin{aligned} \dot{\bar{x}} &= A_{cc}\bar{x} + B_{cc} u \\ \bar{y} &= C_{cc}\bar{x} \end{aligned} \quad (77)$$

Consider the system that has n controllable states. If the maximum number of inputs or outputs is greater than n , then arbitrary n eigenvalues can be assigned. But if the maximum number of inputs or outputs is less than n , then some of the eigenvalues cannot be assigned. There are several ways to assign the eigenstructure. This technique will briefly be reviewed [ref(11)].

Eigenstructure assignment technique with output feedback.

For a linear system given by:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \\ u &= -Ky \end{aligned} \quad (78)$$

where $x \in R^n, u \in R^m, y \in R^r$ the closed-loop regulator is given by

$$\dot{x} = [A - BKC]x \quad (79)$$

and the modes are determined by the eigenvalues of $[A - BKC]$

If the system is completely controllable and observable, and B has full column rank and C has follow row rank, then the following property can be proven [ref(11)]:

- 1) The value of $\max(m,r)$ closed-loop eigenvalue can be arbitrarily assigned
If $n > \max(m,r)$ then the eigenvalue cannot be assigned.
- 2) The shape of the closed-loop eigenvectors associated with the assigned eigenvalues can be assigned as well. A total of $\min(m,r)$ elements of each assigned eigenvector can be chosen arbitrarily.
- 3) The eigenvector associated with the closed-loop eigenvalue \mathbf{I}_i must lie in the subspace spanned by

$$[\mathbf{I}_i \mathbf{I} - \mathbf{A}]^{-1} \mathbf{B} \quad (80)$$

The closed-loop eigenvalue problem looks like:

$$[\mathbf{A} - \mathbf{BKC}] \mathbf{v}_i = \mathbf{I}_i \mathbf{v}_i \quad (81)$$

and we wish to choose K in order to achieve the desired $\mathbf{I}_i, \mathbf{v}_i$ pairs.

The above equation can be rearranged as follows

$$[\mathbf{I}_i \mathbf{I} - \mathbf{A}] \mathbf{v}_i = \mathbf{B} \mathbf{w}_i \quad (82)$$

where $\mathbf{w}_i = -\mathbf{KC} \mathbf{v}_i$

Once the \mathbf{w}_i 's have been found, the gain matrix K is determined by

$$\mathbf{K} = -\mathbf{W}[\mathbf{CV}]^{-1} \quad (83)$$

where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3, \dots, \mathbf{w}_r]$

and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \dots, \mathbf{v}_r]$

This method allows the designer to place r eigenvalues assuming $[CV]$ is invertible.

Eigenstructure assignment technique for full-state feedback case.

For the full-state feedback case, $C = I$ and this implies $r = n$.

This can be considered as 3 separate cases

$m=1$, $1 < m < n$ and $m=n$

1) $m=1$

- a) K has n parameters
- b) All n eigenvalues can be assigned
- c) 1 element of each eigenvector can be assigned (but since eigenvectors have arbitrary scaling in this case it's useless)

2) $m=n$

- a) K has n^2 parameters
- b) All n eigenvalues can be assigned arbitrarily
- c) All element of all eigenvector can be assigned (complete eigenstructure assignment is possible)

3) $1 < m < n$

- a) K has $m \times n < n^2$ parameters.
- b) All n eigenvalues can be arbitrarily assigned
- c) A portion of each eigenvector can be assigned

For each case

$$K = -W[CV]^{-1} \quad (84)$$

where $W = [w_1, w_2, w_3, \dots, w_r]$

and $V = [v_1, v_2, v_3, \dots, v_r]$

If the desired eigenvalues are distinct 'n' values in the assignment process, then the eigenvectors will necessarily be linearly independent and V will be invertible. If the desired eigenvalues are complex eigenvalues then the eigenvectors will also be complex, which result in a complex gain matrix K. To avoid complex gain matrices, the following technique can be used. Consider v_1 and v_2 are complex and conjugate.

$$KV = -W$$

$$\Rightarrow K[v_{1R} + iv_{1I}, v_{1R} - iv_{1I}, v_3, \dots, v_n] = -[w_{1R} + iw_{1I}, w_{1R} - iw_{1I}, w_3, \dots, w_n] \quad (85)$$

Post multiply both sides by

$$\begin{bmatrix} 1/2 & -i/2 & 0 \\ 1/2 & i/2 & 0 \\ 0 & 0 & I \end{bmatrix}$$

which yields:

$$K[v_{1R}, v_{1I}, v_3, \dots, v_n] = -[w_{1R}, w_{1I}, w_3, \dots, w_n] \quad (86)$$

Then $K = -WV^{-1}$, and the resulting K matrix is real.

Thus there is always a real matrix K by simply replacing the complex pair of eigenvectors with the real and imaginary parts of the complex eigenvector.

As described before for the case 2 and 3 the gains are not unique and so the extra freedom can be used to place the components of the eigenvectors.

To do this, the desired eigen pair must satisfy Equation (87).

$$[I_i I - A]v_i = Bw_i \quad (87)$$

where I_i, v_i are the desired eigen pair

Every eigenvector must lie in the subspace spanned by $[\mathbf{I}_i \mathbf{I} - \mathbf{A}]^{-1} \mathbf{B}$. This puts a restriction on the achievable eigenvectors.

Now let's find the algorithm to achieve a vector close to the desired eigenvector if the desired eigenvector v_{id} does not lie in the subspace. To do this let's define a performance index

$$J_i = \frac{1}{2} (v_i - v_{id})^T P_i (v_i - v_{id}) \quad (88)$$

where $v_{id} \equiv$ desired eigenvector

$v_i \equiv$ achievable eigenvector

$P_i \equiv$ a positive definite symmetric matrix whose elements are chosen to weight the difference between v_i and v_{id} for each element.

Using a Lagrange multiplier method,

$$\overline{J}_i = 1/2 (v_i - v_{id})^T P_i (v_i - v_{id}) + \mathbf{u}_i^T ([\mathbf{I}_i \mathbf{I} - \mathbf{A}] v_i - \mathbf{B} w_i) \quad (89)$$

To find the minimum, take partial derivatives with respect to v_i , w_i and \mathbf{u}_i and set to zero.

$$\begin{aligned} \frac{\partial J_i}{\partial v_i} &= P_i v_i + [\mathbf{I}_i \mathbf{I} - \mathbf{A}] \mathbf{u}_i - P_i v_{id} = 0 \\ \frac{\partial J_i}{\partial w_i} &= -\mathbf{B}^T \mathbf{u}_i = 0 \\ \frac{\partial J_i}{\partial \mathbf{u}_i} &= [\mathbf{I}_i \mathbf{I} - \mathbf{A}] v_i - \mathbf{B} w_i = 0 \end{aligned} \quad (90)$$

Which can be written as

$$N_i \begin{Bmatrix} v_i \\ w_i \\ \mathbf{u}_i \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ P_i v_{id} \end{Bmatrix} \quad (91)$$

where

$$N_i = \begin{bmatrix} [I_i I - A] & -B & 0 \\ 0 & 0 & B^T \\ P_i & 0 & [I_i I - A]^* \end{bmatrix} \quad (92)$$

(* denotes hermitian transpose)

It can be shown that as long as the system is completely controllable, N_i will be non-singular.

$$\begin{Bmatrix} v_i \\ w_i \\ \mathbf{u}_i \end{Bmatrix} = N_i^{-1} \begin{Bmatrix} 0 \\ 0 \\ P_i v_{id} \end{Bmatrix} \quad (93)$$

If an eigen pair is not to be altered setting $w_i = 0$ and v_{id} equal to it's original open-loop value assures that the associated eigen pair remain in their open-loop configurations.

So eigenstructure assignment techniques can provide desirable time response characteristics (eigenvalue placement) and modal decoupling (eigenvector specifications) for the nominal plant, but there are no guarantees in terms of stability or performance robustness. This is left to the designer in choosing the desired eigenstructure.

State-space model with limited measurement data.

Suppose we have \bar{A} , which has an eigenvalue set $\Lambda = \{\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n\}$

Then the characteristic polynomial for matrix \bar{A} is

$$\begin{aligned} |sI - \bar{A}| &= (s - \lambda_1)(s - \lambda_2)(s - \lambda_3) \dots (s - \lambda_n) \\ &= s^n + a_1 s^{n-1} + a_2 s^{n-2} \dots + a_n \end{aligned} \quad (94)$$

If \bar{A} represents closed-loop system A matrix which is $A - B^*K^*C$, then the equation (94) can be described as follows:

$$|sI - \bar{A}| = \sum_{g=0}^n f(k_{ij}) s^g \quad (95)$$

where k_{ij} represents the i^{th} row and j^{th} column element of regulator gain matrix K.

If the desired closed-loop poles are specified as eigenvalue set Λ then the characteristic polynomial for matrix \bar{A} must satisfy the equation (94) and (95).

By comparing the coefficient of the characteristic polynomial between equation (94) and (95), every element of the regulator gain matrix K can be found. This method can be used for the multi-input-multi-output case too.

In single-input-single-output case reference (2) page 833~834 shows an easy way determining the K matrix. It should be also noted that in the SISO case the K matrix is unique. But in the multi-input-multi-output case, the K matrix is not unique. There are several ways to choose the K matrix.

As developed in Chapter 2, the model has 4 inputs, 4 outputs and 12 states. The model output information is only for the displacement states. The dynamic system can be described as follows:

$$\begin{aligned} \begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix} &= A \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + Bu = \begin{bmatrix} 0 & I \\ -Ks & -Cd \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + \begin{bmatrix} 0 \\ Bs \end{bmatrix} u \\ Y &= C \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + Du = [C_m \quad 0] \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \end{aligned} \quad (96)$$

where x represents the displacement state vector and \dot{x} the velocity. If the system is strictly proper, then the D matrix can be removed. Here Ks represents normalized stiffness matrix, Cd represents normalized damping matrix, Bs represents input direction for the forcing element and C_m represents output.

The open-loop system characteristic equation is:

$$\det[A - II] = \begin{vmatrix} -II & I \\ -Ks & -Cd - II \end{vmatrix} = 0 \quad (97)$$

The coefficients of I^n and I^{n-1} (the two highest polynomial terms) happen only at the product of the diagonal term.

Now let's assume that we can measure only displacement.

$$y = [C_m \quad 0] \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \quad (98)$$

With this open-loop system let's build the closed-loop system with output feedback.

$$u = -Ky \quad (99)$$

Then the system will be:

$$\begin{aligned}
 x &= \begin{bmatrix} 0 & I \\ -Ks & -Cd \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} - \begin{bmatrix} 0 \\ Bs \end{bmatrix} K \begin{bmatrix} C_m & 0 \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \\
 &= \begin{bmatrix} 0 & I \\ -Ks - BsKC_m & -Cd \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \\
 &= A_{cl} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix}
 \end{aligned} \tag{100}$$

where

$$A_{cl} = \begin{bmatrix} 0 & I \\ -Ks - BsKC_m & -Cd \end{bmatrix} \tag{101}$$

The characteristic equation of the closed-loop system A_{cl} matrix is:

$$\det[A_{cl} - II] = \begin{vmatrix} -II & I \\ -Ks - BsKC_m & -Cd - II \end{vmatrix} = 0 \tag{102}$$

Again, the coefficients of I^n and I^{n-1} happen only at the product of the diagonal term. But in this product any control element of K matrix cannot affect the coefficient of the highest and the next highest term of I . That means that the coefficient of I^{n-1} is fixed. But as noticed the coefficient of I^{n-1} for the closed-loop system is the same as the open-loop system. In calculating the polynomial, the coefficient of the I^{n-1} term of the characteristic equation is equal to the sum of all eigenvalues. If one of the desired eigenvalues is relatively larger than before, then the other eigenvalue should be relatively smaller and the sum of eigenvalues should be the same as the open-loop eigenvalue sum. If the controller is to stabilize a mode extremely fast, i.e. the desired eigenvalue's real value is extremely large with negative sign, then at least one of the other eigenvalue's

real value can be positive, i.e. the closed-loop system could become unstable. So in the eigenstructure assignment, there is an inherent trade-off of the desired eigenvalues.

We need to measure the output velocity in order to guarantee arbitrary eigenvalue assignment. The results shown in Chapter 4 used $\begin{bmatrix} C_m & 0 \\ 0 & C_m \end{bmatrix}$ as the output measurement matrix C , to guarantee the placement of desired eigenvalue. In practice, this could be accomplished with both displacement and velocity sensors or with velocity sensors and digital integrators for the displacement.

Determining the desired eigenvalues

From a given open-loop system, the related frequency response can be found by using a singular value plot. The peak points represent the output strength with the related natural frequency. The objective of this thesis is to minimize the output variance for a random input. If that output variance is zero when the input is random (assumed zero-mean White Gaussian Noise), that is the best result. But it is impossible to make the output zero, but minimizing the output variance, the closed-loop system will maintain focus better than the open-loop system.

The output variance can be changed by adjusting the peak singular values with related frequency in open-loop system. If the peak singular value is lowered, the output variance can be lowered. The eigenstructure assignment technique can be used to lower the peak singular values. (which are directly related to the eigenvalues)

The idea behind selecting the lowering of the peak singular values is as follows. Figure 10 shows the relations between real, imaginary and absolute value of eigenvalue, where the open-loop eigenvalue is $\mathbf{I}_i = -a + bi$.

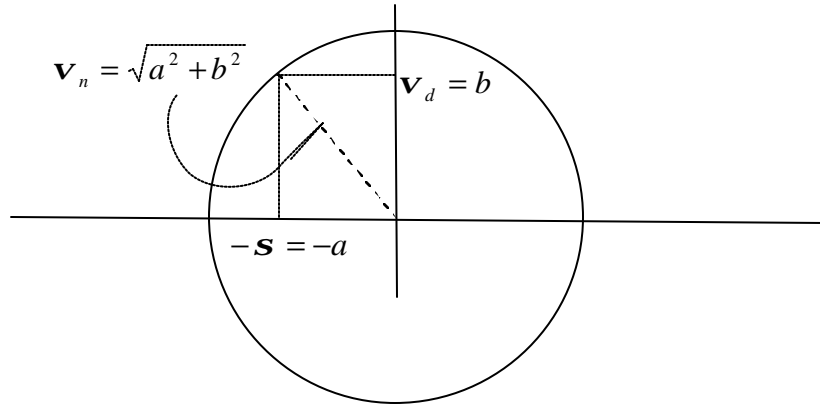


Figure 10. Relation between Natural frequency and Damping frequency

The speed of stabilizing the system strictly depends on the real part of the eigenvalue which is called an attenuation constant. So minimizing the output variance, or lowering the peak singular value, is maximizing the eigenvalue's real part 'a' which maintaining the same natural frequency.

But each singular value does not have the same co-relation with the output variance. The output variance is dominated by the lower frequency eigenvalues because the modes with lower natural frequencies decay slower with respect to higher natural frequency modes. So the modes which occur at the lowest frequencies and have high singular values are the best candidate modes to lower their peak singular values.

But selecting the best real eigenvalue must satisfy $\mathbf{s}^2 \leq \mathbf{v}_n^2$ and the constraint of the max allowable gain value. The control effort required, which is directly related to the required gain, is dependant on the required changes to the eigenvalue. Thus these gain increases were minimized by increasing the real part but maintaining the frequency at which the modes occur [ref(11)].

Determining the desired eigenvector

As described in the section Eigenstructure Assignment technique with output feedback, the desired eigenvector associated with the closed-loop eigenvalue λ_i must lie in the subspace spanned by:

$$V_{i-1} = [I_i I - A]^{-1} B \quad (103)$$

This is the first necessary condition of the desired eigenvector.

In the full-state feedback case, or a totally controllable system, the condition stated above of the desired eigenvector is always satisfied, but a system which has rigid body modes or uncontrollable modes is not.

As described in the section on Analyzing the open-loop Eigenstructure, every closed-loop mode shape which is controllable lies in the subspace spanned by the open-loop controllable eigenvectors (mode shapes). Let M_{OL} be a basis of the set of open-loop mode shapes. Then the closed-loop mode shape M_{CL} can be determined by a linear combination of open-loop mode shape.

$$\begin{aligned} M_{CL} &= M_{OL} \times \{c\} \\ \{c\} &= \{a_1 \quad a_2 \quad \dots \quad a_k\}^T \end{aligned} \quad (104)$$

where k is the number of linearly independent open-loop mode shapes.

Then the closed-loop eigenvector V_{CL} with arbitrary eigenvalue must be:

$$V_{CL} = \begin{bmatrix} M_{CL} \\ I \times M_{CL} \end{bmatrix} = \begin{bmatrix} M_{OL} \times \{c\} \\ I \times M_{OL} \times \{c\} \end{bmatrix} = \begin{bmatrix} M_{OL} \\ I \times M_{OL} \end{bmatrix} \{c\} \quad (105)$$

So any closed-loop eigenvector related with eigenvalue λ_i must lie in the subspace spanned by $\begin{bmatrix} M_{OL} \\ I \times M_{OL} \end{bmatrix}$ and let's call that as V_{i_2} . This is the second necessary condition of the desired eigenvector.

So desired eigenvector must satisfy both necessary conditions. But there is an intersecting space which satisfies the first and second necessary condition as shown in Figure 11. It is necessary to use the intersection space in finding the desired eigenvector.

The intersection space must lie in the subspace of the first and second necessary condition. If $\{x\}$ is a vector which lies in the intersection space of the two necessary conditions space, then the vector $\{x\}$ can be expressed such as:

$$\{x\} = [V_{i_1}]\{a\} = [V_{i_2}]\{b\} \quad (106)$$

where $\{a\}$ and $\{b\}$ represents linear combination of column space of the intersection of matrix V_{i_1} and V_{i_2} .

i.e.

$$[V_{i_1}]\{a\} - [V_{i_2}]\{b\} = \{0\} \quad (107)$$

$$\text{so } [V_{i_1} \quad V_{i_2}] \begin{Bmatrix} a \\ -b \end{Bmatrix} = \{0\} \quad (108)$$

Therefore vector $\begin{Bmatrix} a \\ -b \end{Bmatrix}$ is null space of $[V_{i_1} \quad V_{i_2}]$.

So the intersection space (V) of the two necessary condition spaces is $\text{span}\{[V_{i_1}]\{a\}\}$ or $\text{span}\{[V_{i_2}]\{b\}\}$. The space V represents the achievable desired eigenvector space.

$$\begin{aligned} V &= \text{span}\{[V_{i_1}]\{a\}\} \\ &= \text{span}\{[V_{i_2}]\{b\}\} \end{aligned} \quad (109)$$

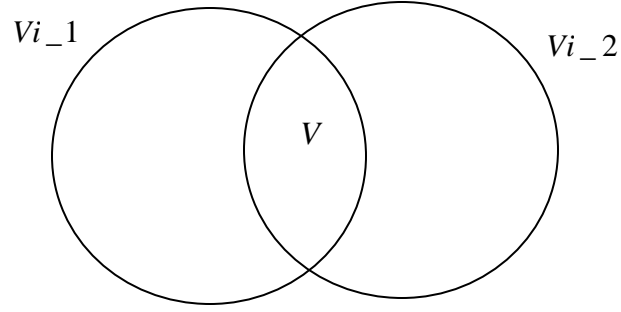


Figure 11. Necessary Condition Set Relationship Diagram

Then the best desired eigenvector (\mathbf{n}_{id}) is a vector which lies in the space of V and minimizes the ' J ' as in following equation:

$$\begin{aligned} Y_{ni} &= C\mathbf{n}_{id} \\ J &= Y_{ni}^T \times P \times Y_{ni} \end{aligned} \quad (110)$$

where P is any positive semi-definite penalty matrix and \mathbf{n}_{id} is a vector lies in the space V .

Physically, equation (110) tells us to select eigenvectors that contribute to the weighted cost ' J '.

In reality as the eigenvalue moves, the related eigenvector moves from open-loop eigenvector to a vector which lies in the space of achievable desired eigenvector (V). But as eigen-pair (eigenstructure) approaches the desired eigenvalue and the best desired eigenvector, the elements of control gain matrix increases. The allowable maximum control gain constraints limit the extent to which the desired eigenvectors can be achieved. In this thesis, the proper desired eigenstructure which satisfies allowable maximum control gain constraints was found by using Newton's Line Search Method. Typically a desired eigenvector was found by finding the proper ratio value r_i in Equation (111) between 1 through 0.

$$\mathbf{n}_{ia} = \mathbf{n}_{io} r_i + (1 - r_i) \mathbf{n}_{id} \quad (111)$$

where \mathbf{n}_{ia} is the i^{th} mode assigned eigenvector as the desired one, \mathbf{n}_{io} is the i^{th} mode open-loop system eigenvector, and \mathbf{n}_{id} is i^{th} mode best desired eigenvector.

Every assigned eigenvector was found by finding related r_i simultaneously using the Newton's Line Search Method again.

For comparison with the eigenstructure assignment technique, an LQR approach was also used, and will briefly be discussed below.

Linear Quadratic Regulator(LQR).

LQR is the well known in finding optimal regulator. As shown in Equation (112), the LQR problem is stated as determine $u(t)$ which minimize performance index:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (112)$$

where Q is positive semi definite and R is positive definite matrix.

The solution which will determine the control law $u(t)$ and hence compensator is found from calculus of variation techniques. Using a Lagrange multiplier to convert this constrained n^{th} order minimization to $2n^{\text{th}}$ order unconstrained minimization.

In the steady state case, LQR problem turns out to be the solution to the Algebraic Riccati Equation (ARE). The ARE is solved in MATLAB. This thesis used LQR command in MATLAB by assigning the diagonal value of matrix Q and R .

Summary

This chapter established the technique for achieving the proper gains for the eigenstructure assignment technique and showed the simple approach with LQR method.

In using the eigenstructure assignment technique, the number of measurement outputs was an important factor in achieving desired eigenvalues. If there are not enough measurement outputs rather than states, restricted eigenvalue could be assigned. To overcome this restriction, a proper state estimator or more output measurement was needed. With the assumption of existence of best estimator, eigenstructure assignment can be fully developed.

The next chapter shows the result using eigenstructure assignment technique. The result of using LQR is also shown to compare the result from eigenstructure assignment technique. Finally, the AFRL model analysis results are added at the end.

IV. Result and Analysis

The previous chapter developed eigenstructure assignment and LQR methods. This chapter shows the result of the 2-D planar model with both techniques and the AFRL model analysis result using the eigenstructure assignment technique.

Open-loop system analysis (2-D planar Model)

Assumptions:

- 1). There is no output measurement error
- 2). In developing the EOMS (Equations of motion), small angle and small displacement approximations were used to make the control system linear
- 3). Both displacements and velocities can be measured.

Constraints:

- 1). Control gain max value of 100 (max absolute value of any element of 'K' matrix).

As shown in Chapter 1 problem statement, the given system can be described as follows:

$$\overline{M}\ddot{x} + \overline{C}\dot{x} + \overline{K}x = \overline{D}u \quad (1)$$

where

\overline{M} : Mass matrix

\overline{C} : Damping matrix

\bar{K} : Stiffness matrix

For the EOM above, the state-space system can be developed (with the assumption of output velocity measurements) as follows:

$$\begin{aligned} \begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix} &= A \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + Bu = \begin{bmatrix} 0 & I \\ -\bar{M}^{-1}\bar{K} & -\bar{M}^{-1}\bar{C} \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + \begin{bmatrix} 0 \\ \bar{D} \end{bmatrix} u \\ \begin{Bmatrix} y \\ \dot{y} \end{Bmatrix} &= C \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} = \begin{bmatrix} C_m & 0 \\ 0 & C_m \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \end{aligned} \quad (78)$$

where C_m represents measurement matrix for state the x .

In this thesis, parameters in the \bar{M} , \bar{K} and \bar{C} were chosen to get similar characteristics of performance of the AFRL DOT model system. The AFRL's DOT

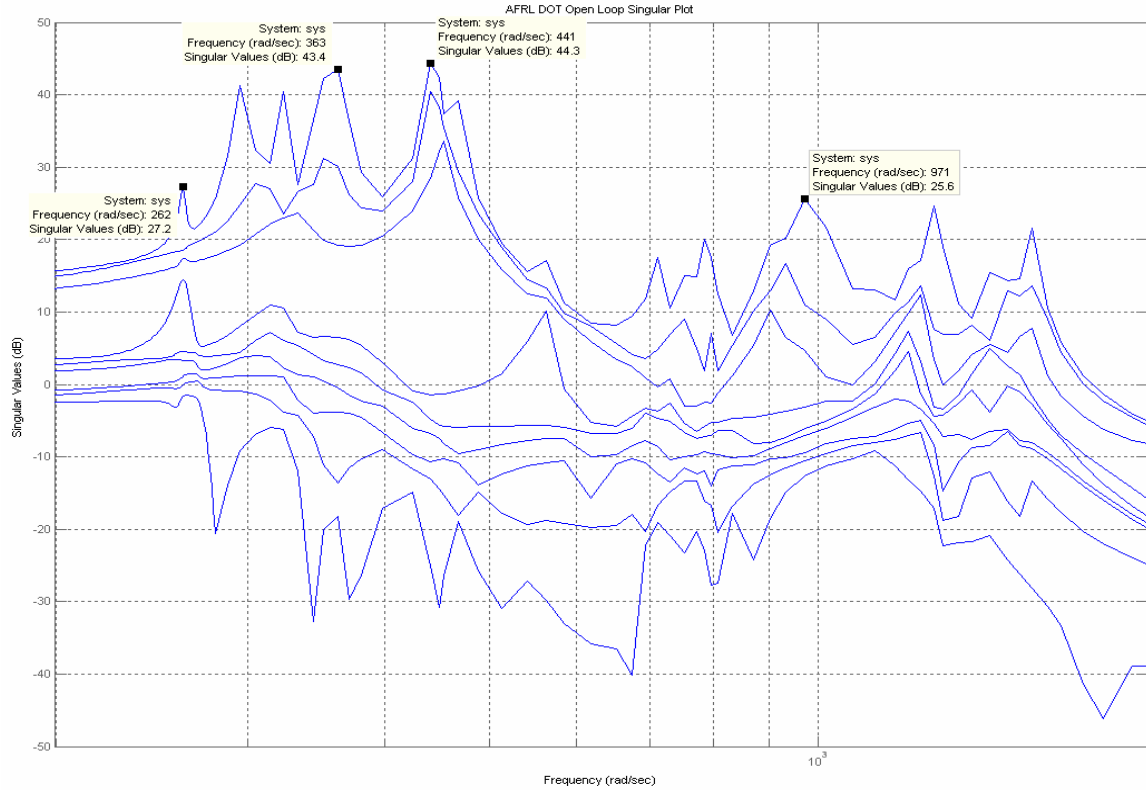


Figure 12. Open-Loop System Singular Plot (AFRL DOT System)

model singular value plot is shown in Figure 12, and the planar model singular value plot is shown in Figure 13.

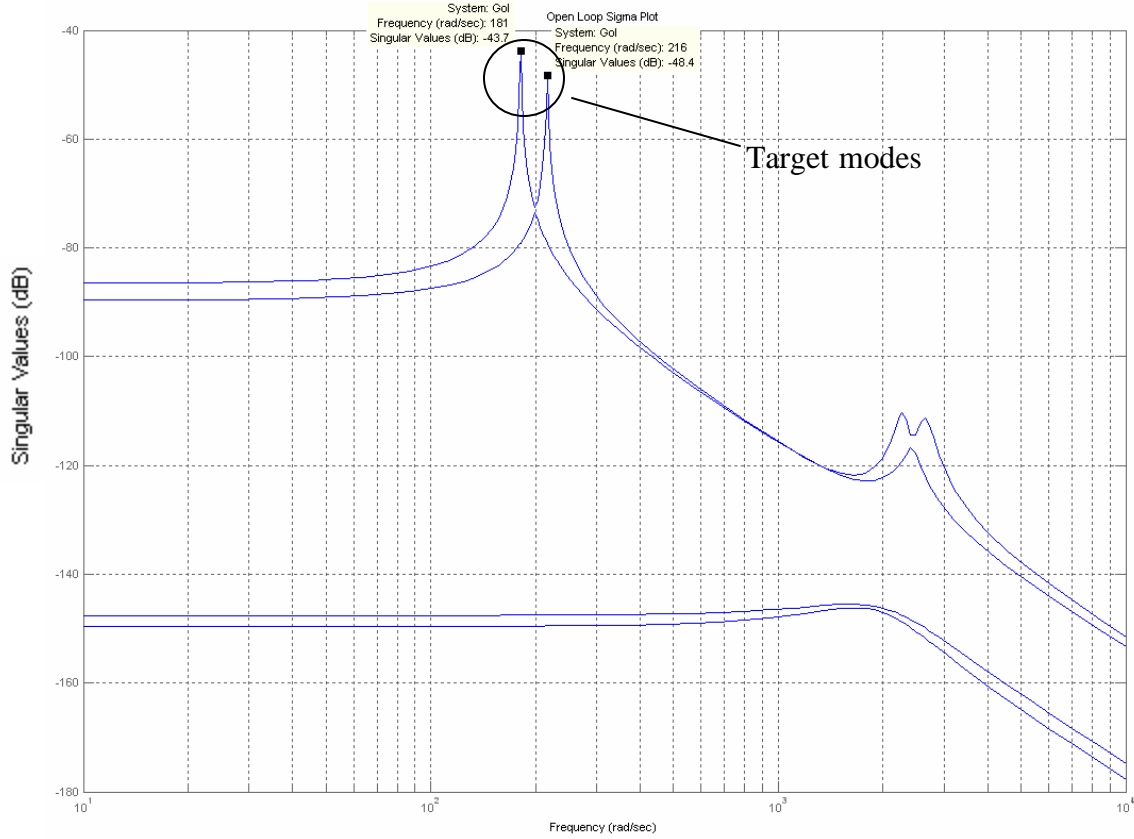


Figure 13. Open-Loop System Singular Plot (2-D Planar Model)

Table 1. Open-Loop System Mode

		1st mode	2nd mode	3rd mode	4th mode	5th mode	6th mode
Eigenvalue ($\times 10^3$)		-0.1401 $\pm 2.6431i$	-0.1040 $2.2780i$	-0.0007 $\pm 0.1812i$	-0.0009 $\pm 0.2165i$	0	0
Damping		5.29e-002	4.56e-002	3.62e-003	4.33e-003		
Mode Shape	z_1	0.6803	0.7068	0	0.2699	0.5774	-
	z_2	-0.6803	0.7068	0	-0.2699	0.5774	0.4472
	z_b	0	-0.0283	0	0	0.5774	0.4472
	θ_1	-0.0013	0	0.7071	0.6256	0	0
	θ_2	-0.0013	0	-0.7071	0.6256	0	0.4472
	θ_b	0.2729	0	0	-0.2674	0	0.4472
Controllability		Controllable				Uncontrollable (Rigid body mode)	

The open-loop mode shapes were determined by finding the eigenvalues and eigenvectors. Each mode's eigenstructure is shown in Table 1 and each mode shape is shown in Figure 14.

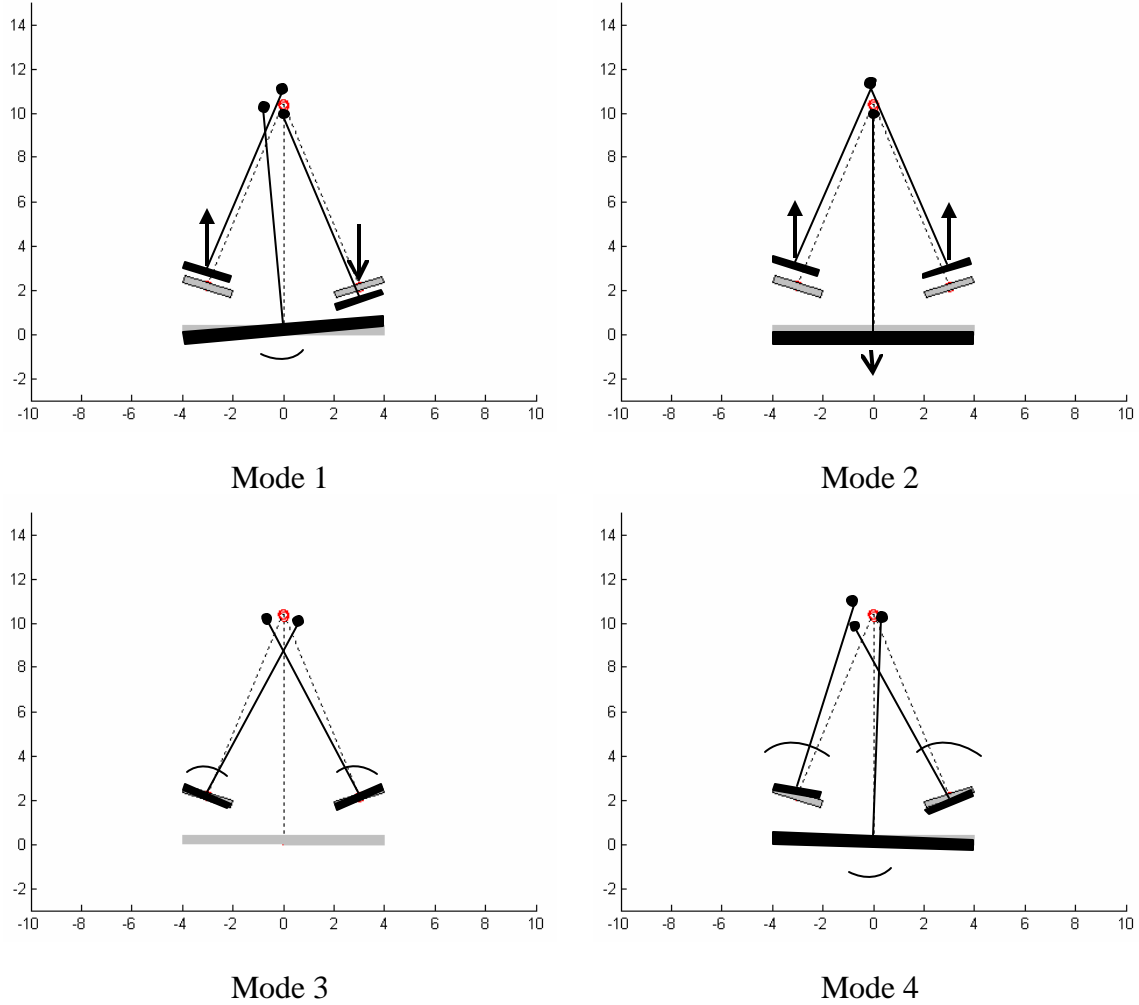


Figure 14. Open-Loop System Mode Shape (2-D Planar Model)

As assumed from the beginning, the satellite system can have several disturbance sources (noise). All the sources of noise were assumed to be White Gaussian Noise with intensity 2×10^{-7} to be close to the AFRL model which had nano range disturbances.

Using the Lyapunov equation, the steady-state output variance and Root Mean Square (RMS) value was computed. There are 4 displacements and angle errors and 4 velocity errors. The velocity error variance was not used in the objective function. Intuitively, the tilt angle errors are more important than the displacement errors, so the objective function used was as follows:

$$J = R^T \times P \times R \quad (113)$$

where

R : RMS of $\{T_1 \quad T_2 \quad D_1 \quad D_2\}^T$

T_1 : Tilt Angle Error for Mirror 1

T_2 : Tilt Angle Error for Mirror 2

D_1 : Displacement Error for Mirror 1

D_2 : Displacement Error for Mirror 2

P : penalty positive semi-definite matrix

(in this thesis $P = \text{diag}(10^4 \quad 10^4 \quad 1 \quad 1)$)

The open-loop RMS results are provided in Table 2. The result of objective function was 4.1718×10^{-10} .

Table 2. Open-Loop System Measurement RMS (2-D Planar Model)

	Planar Model ($\times 10^{-9}$)		AFRL Model ($\times 10^{-9}$)	
Mirror “1”	Tilt Error	143	143.3	A segment X-tilt
	Piston Error	2028.9	46.68	A segment piston
Mirror “2”	Tilt Error	143	170.5	C segment X-tilt
	Piston Error	2028.9	60.58	C segment piston

Through analyzing the open-loop system singular value plot, the candidate modes to be changed were determined as the mode which has largest singular value and minimum absolute real eigenvalue.

As shown in Figure 13, there are two candidate modes to be changed. At first, the objective function can be reduced by selecting only one mode and increasing the real part of the eigenvalue which satisfies the constraint that the maximum control gain matrix element must be less than allowed value. The constraint used was 100.

Closed-loop system assigned 3rd mode eigenvalue

The 3rd mode eigenvalue $-0.657 \pm 181i$ has the smallest real value, i.e. the slowest mode. So the assumption that the 3rd mode has a large contribution to the objective function result was made. By assigning a different eigenvalue real part and the same eigenvector with open-loop system at the same frequency, the objective function result can be lowered. But as the eigenvalue real part is increased, the control gain matrix element value was increased, as expected. Thus, there was a unique eigenvalue that satisfies the constraint that the maximum the control gain matrix element value was less than 100 (actually it was equal to 100).

By assigning the maximum control gain matrix element value as 100, the resulting singular value plot with respect to frequency was given in Figure 15.

The resulting eigenvalue difference is shown in Table 3.

Table 3. 3rd Mode Eigenvalue Comparison.

	Open Loop	Closed Loop
Eigenvalue	$-0.657 \pm 181i$	$-27.1 \pm 179i$
Damping	3.62×10^{-3}	1.50×10^{-1}

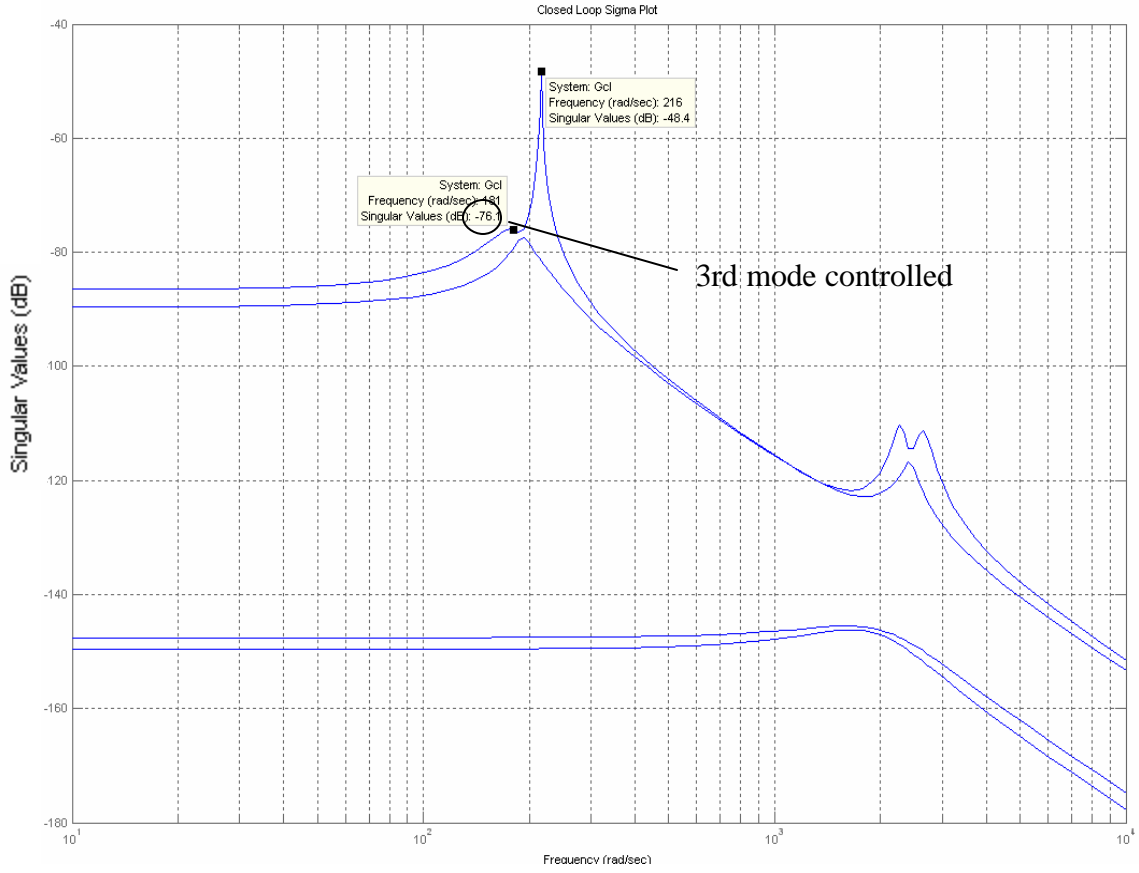


Figure 15. Closed-Loop System Singular Plot (3rd mode changed)

The result of objective function was 2.1367×10^{-10} which was almost 51.2 % of open-loop result. The resulting jittering performance is shown in Table 4.

Table 4. Closed-loop System Measurement RMS (3rd mode changed)

($\times 10^{-9}$)		Open-Loop Planar Model	Closed-Loop Planar Model
Mirror “1”	Tilt Error	143	103
	Piston Error	2028.9	1193
Mirror “2”	Tilt Error	143	103
	Piston Error	2028.9	1193

Closed-loop system assigned 4th mode eigenvalue

Using the same procedure, the 4th mode eigenvalue ($-0.937 \pm 216i$) has as small eigenvalue real part as 3rd mode. That means that the 4th mode still has a contribution to the objective function result as much as the 3rd mode. Then by assigning a different eigenvalue real part and the same eigenvector as the open-loop system at the same frequency, the objective function result was lowered to 2.1469×10^{-10} which is 51.5 % of open-loop result.

Figure 16 shows the resulting singular value plot with respect to frequency with the same max allowable control gain matrix element value constraint. The resulting eigenvalue differences are shown in Table 5.

Table 5. 4th Mode Eigenvalue Comparison.

	Open Loop	Closed Loop
Eigenvalue	$-0.937 \pm 216i$	$-32.5 \pm 214i$
Damping	4.33×10^{-3}	1.50×10^{-1}

Table 6. Closed-Loop System Measurement RMS (4th mode changed)

$(\times 10^{-9})$		Open-Loop Planar Model	Closed-Loop Planar Model
Mirror "1"	Tilt Error	143	102
	Piston Error	2028.9	1673
Mirror "2"	Tilt Error	143	102
	Piston Error	2028.9	1673

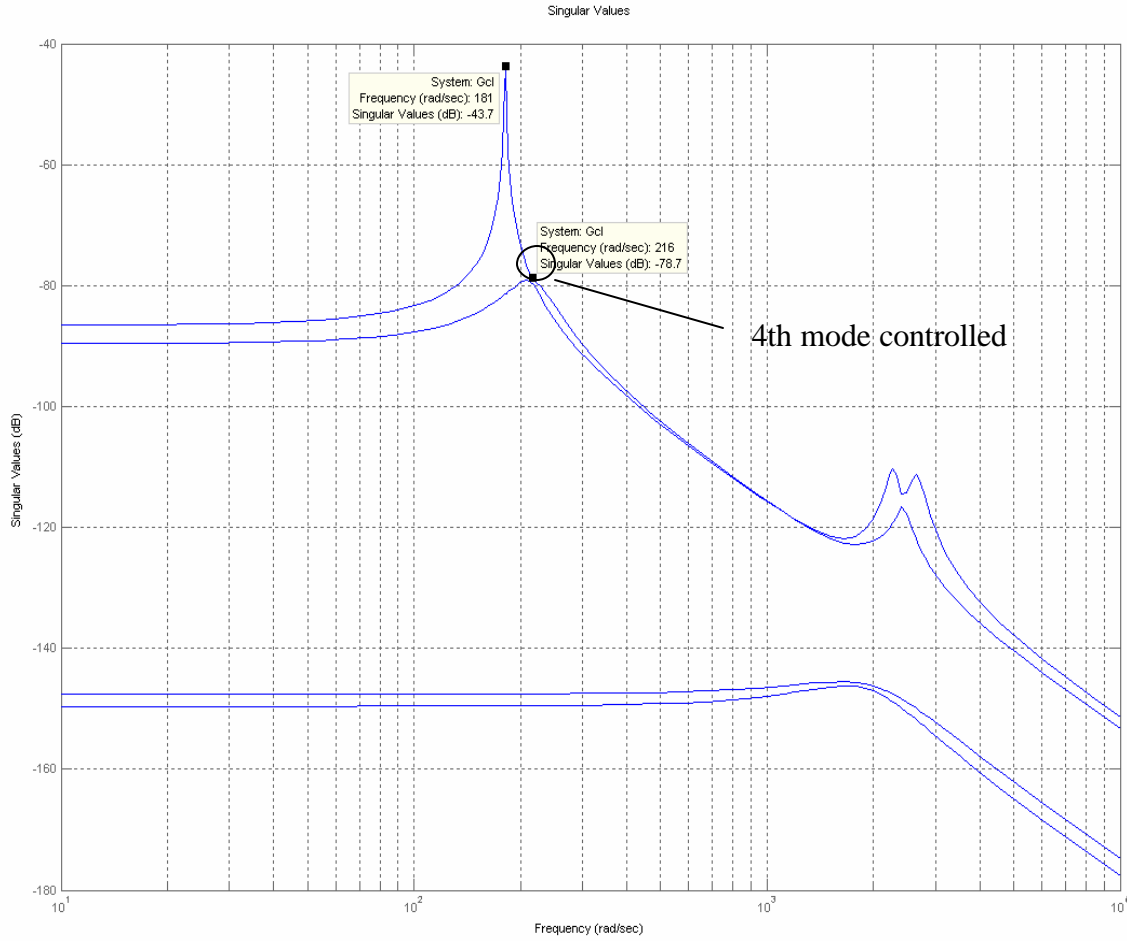


Figure 16. Closed-Loop System Singular Plot (4th mode changed)

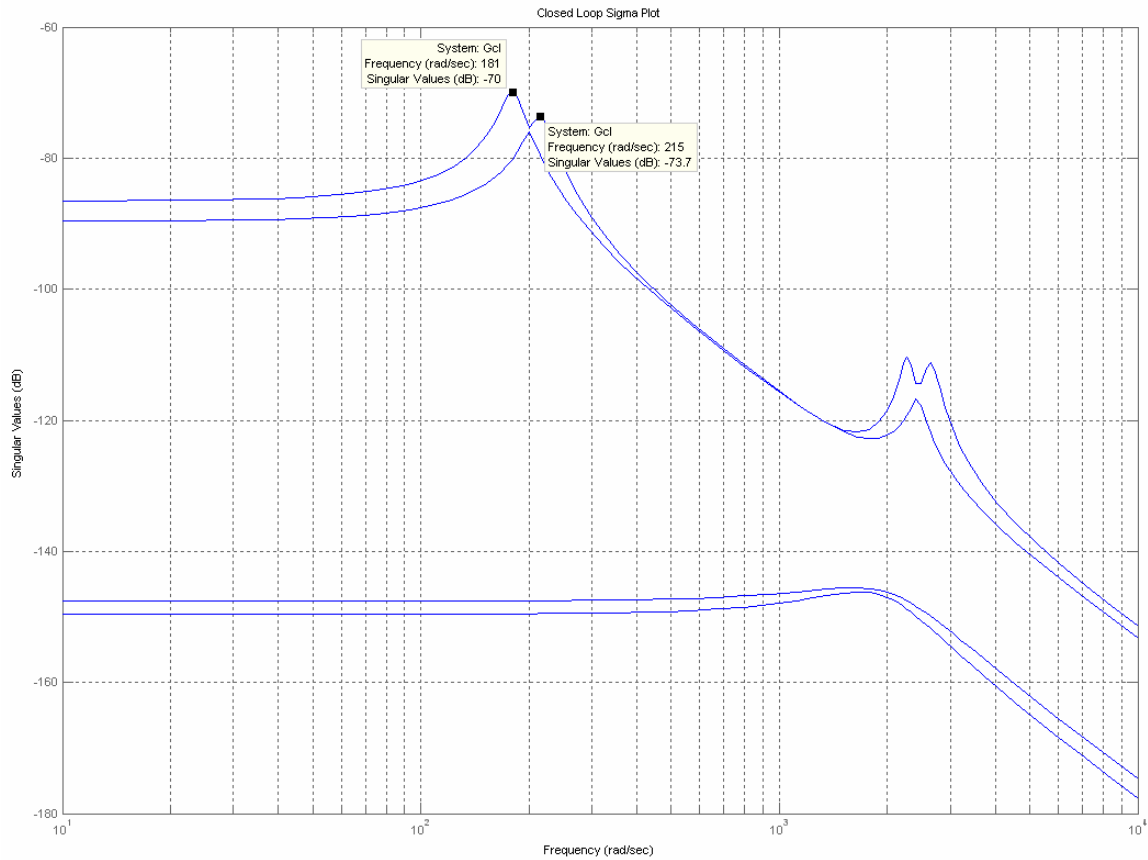
Closed-loop system assigned 3rd and 4th mode eigenvalue

The previous section changed the eigenvalue only for a single mode and showed the possible eigenvalue range that can be achieved. So the assumption that the objective function result could be lowered by changing two eigenvalues at the same time in the achievable range was made. Newton's Line Search Method which is developed in Matlab as built-in function with command name of '*fmincon*' was used in finding the best two eigenvalue pair to minimize.

Table 7 and Figure 17 show the optimized result only by changing the eigenvalues, but with the same open-loop system eigenvectors.

Table 7. 3rd and 4th Mode Eigenvalue Comparison.

	Open Loop		Closed Loop	
	3 rd mode	4 th mode	3 rd mode	4 th mode
Eigenvalue	$-0.657 \pm 181i$	$-0.937 \pm 216i$	$-13.3 \pm 181i$	$-17.4 \pm 216i$
Damping	3.62×10^{-3}	4.33×10^{-3}	7.36×10^{-2}	8.02×10^{-2}



**Figure 17. Closed-Loop System Singular Plot
(3rd and 4th mode changed at the same time)**

Table 8. Closed-Loop System Measurement RMS (3rd and 4th mode changed)

($\times 10^{-9}$)		Open-Loop Planar Model	Closed-Loop Planar Model
Mirror “1”	Tilt Error	143	32.6
	Piston Error	2028.9	457.5
Mirror “2”	Tilt Error	143	32.6
	Piston Error	2028.9	457.5

After changing only the eigenvalues for modes 3 and 4, the objective function result was lowered to 0.2164×10^{-10} which is 5.2% of the open-loop result.

Closed-loop system assigned 3rd and 4th mode eigenvector

With the same eigenvalues achieved above, there can exist desired eigenvectors to further minimize the objective function.

Using the technique from Chapter 3 Equation (109) and (110), the desired eigenvectors were determined and the related mode shape is shown in Table 9.

Table 9. 3rd and 4th Mode Optimum Mode Shape.

	Mode Shape			
	3 rd Mode		4 th Mode	
	Open Loop	Desired	Open Loop	Desired
z_1	0	-0.7054	0.2699	-0.7054
z_2	0	-0.7054	-0.2699	-0.7054
z_b	0	0.0282	0	0.0282
θ_1	0.7071	0.0456	0.6256	0.0456
θ_2	-0.7071	-0.0456	0.6256	-0.0456
θ_b	0	0	-0.2674	0

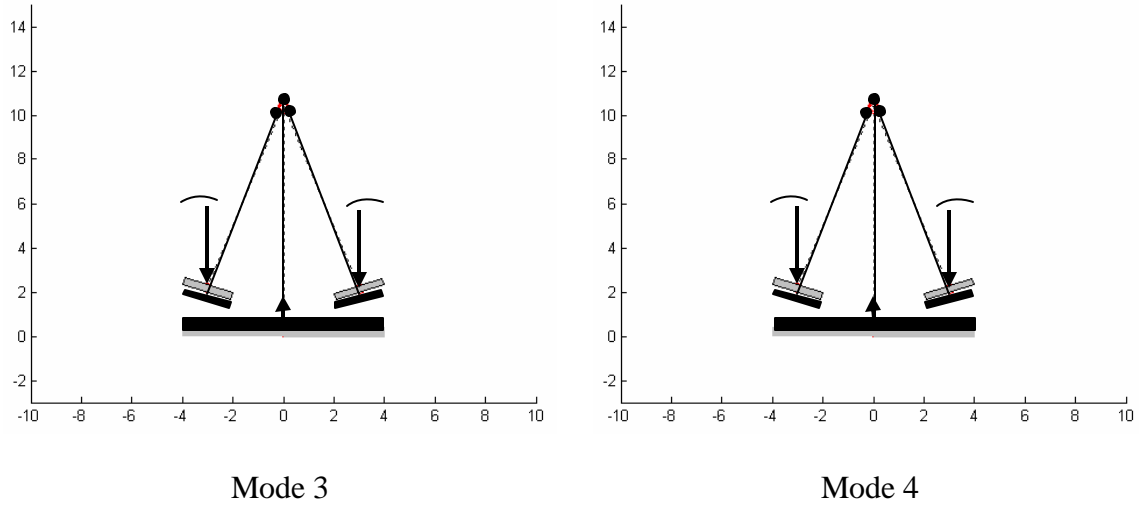


Figure 18. Desired Closed-Loop Mode Shape (2-D Planar Model)

As a result, using the desired eigenvector with the mode shape as described in Table 9, the eigenvector was exactly achieved and the resulting objective function was lowered to 0.1244×10^{-10} which is 2.98 % of open-loop result. But the maximum gain matrix element was increased to 9.99×10^6 . So by assigning the desired eigenvector between open-loop eigenvector to the best achievable eigenvector as described in Equation (113) in Chapter 3, the maximum gain matrix element could be lowered to 100 and the achieved result of objective function was 0.2164×10^{-10} which is almost the same result of the previous section. The achieved closed-loop mode shape was almost same with the result of previous section which was similar to the open-loop mode shape.

The results showed that using the desired eigenvectors can further reduce the objective function, but at an increase in gain and hence control usage. The technique allows the designer to quickly adjust the amount of ‘shape control’ achieved (eigenvector) for a given amount of control use.

Closed-loop control using LQR

The advantage of LQR is the guaranteed stability with plenty of margins. The cost function minimized was Equation (112) Chapter 3, repeated here for ease of reading.

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (112)$$

The given system in this thesis has uncontrollable rigid body modes. One of the requirements in using LQR is that the system must be completely controllable. So only the controllable modes were used by separating the system into controllable and uncontrollable subsystem as outlined in Chapter 3. The LQR problem can be converted to a problem of finding the proper penalty value for the state and control usage. To minimize the search space, this thesis assumed that all the control usage has the same importance (i.e. all the actuators are equally weighted). So every diagonal value of the matrix R was set to 0.1. With this, the problem is now finding the proper diagonal value of the Q matrix. To find the proper diagonal value, Newton's Line Search Method was used again.

As a result, the 1st and 2nd mode eigenvalues were almost the same as the open-loop system. The 3rd and 4th mode eigenvalues were changed mostly, but the values were slightly different from the result of assigning the 3rd and 4th mode eigenvalues.

The resulting singular plot and the eigenvalue comparison is given in Figure 16 and in Table 10.

The resulting objective function was lowered to 0.2169×10^{-10} which is 5.2 % of open-loop result (same as the eigenstructure assignment technique).

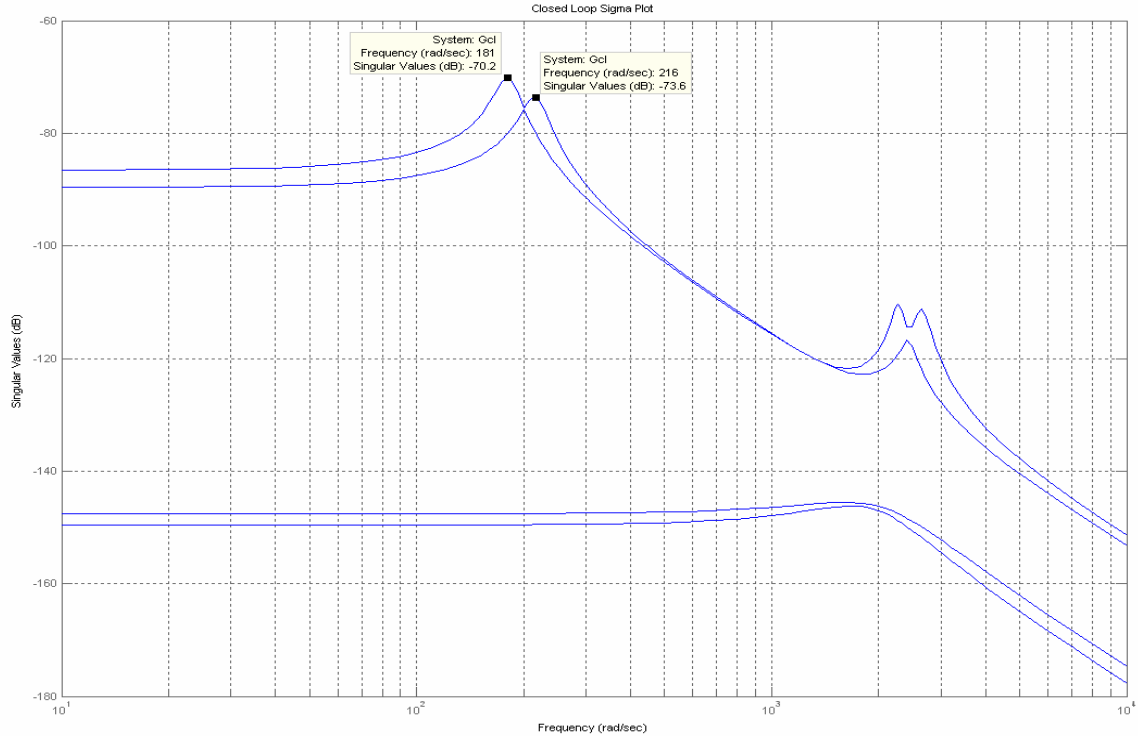


Figure 19. Closed-loop System Singular Plot (Using LQR)

Table 10. 3rd and 4th Mode Eigenvalue Comparison(Using LQR)

	Open Loop		Closed Loop	
	3 rd mode	4 th mode	3 rd mode	4 th mode
Eigenvalue	$-0.657 \pm 181i$	$-0.937 \pm 216i$	$-13.7 \pm 181i$	$-16.9 \pm 216i$
Damping	3.62×10^{-3}	4.33×10^{-3}	7.54×10^{-2}	7.82×10^{-2}

Table 11. Closed-Loop System Measurement RMS (Using LQR)

$(\times 10^{-9})$		Open-Loop Planar Model	Closed-Loop Planar Model
Mirror “1”	Tilt Error	143	32.6
	Piston Error	2028.9	456
Mirror “2”	Tilt Error	143	32.6
	Piston Error	2028.9	456

Comparison between Eigenstructure assignment technique and LQR approach

In many cases LQR was used because LQR guarantees the stability if it is possible to measure the output exactly. The number of penalty variables to be determined using LQR was equal to the sum of the number of states and the number of inputs. In this thesis, the simplified model had 8 penalty variables and it was the same as the number of system eigenvalues. But the eigenstructure assignment technique used several eigenvalues (or modes) which have small real eigenvalue parts and mostly occur in the low frequency band. The number of target eigenvalues was less than number of states and should be less than half because of the complex conjugate stipulation. In this thesis there was 2 eigenvalues and their complex conjugates, was used as the target mode eigenvalue. The resulting objective function value after using the eigenvalue assignment was pretty much equivalent with the result of using the LQR method. So assigning the eigenvalue was a computationally faster method than LQR.

Eigenvector assignment was effective to reduce the objective function but most of the reduction is attributed to the eigenvalue assignment. Finding the optimum eigenvectors (optimum mass motion) was the same as finding the linear combination from the first and second desired eigenvector necessary conditions. Achieving the optimized eigenvectors results in a requirement of high control gain values, such as an order of 10^9 in the simplified model. After finding and assigning the best desired eigenvector between open-loop system eigenvector and optimized eigenvector by Newton's Line Search Method (Equation (111) in Chapter 3), resulting objective function was similar to the result from assigning just the eigenvalues.

Closed-loop system for the AFRL DOT system with Eigenvalue Assignment

The actual DOT system model which AFRL developed had 9 inputs and 9 outputs and 79 states. The corresponding open-loop singular value plot and optical jitter output was shown previously Figure 12 and is tabulated in Table 12.

To reduce the computation time, this thesis used a balanced model realization and model reduction to 20 states, with 9 inputs and 9 outputs. The obtained singular value plot is shown in Figure 20. After reducing the system, a combination of noise intensities was assumed for the reduced model open-loop system to be close to the original system's jitter performance.

The assumed noise intensity and resulting open-loop performance is shown in Tables 13 and 14.

The objective function was determined as the sum of output variance and the open-loop system result was 2.3629×10^{-11} .

Table 12. Optical Performance (AFRL DOT System)

Segment	D.O.F.	Open-Loop Jitter	Specification	Units (rms)
A	piston	46.68	14	nm
	X-tilt	168.44	95	nrad
	X-tilt	143.03	95	nrad
B	piston	45.64	14	nm
	X-tilt	106.57	95	nrad
	X-tilt	89.96	95	nrad
C	piston	60.58	14	nm
	X-tilt	170.5	95	nrad
	X-tilt	117.88	95	nrad

In using the eigenstructure assignment technique, a full state feedback system or a system which has the same number of output measurements as the number of states guarantees every desired eigenvalue can be achieved. The system that AFRL had developed has only 9 inputs and 9 outputs. This is less than the number of states. This means the eigenvalue assignment technique has a restriction (trade-off) as discussed in Chapter 3 (State-space model with limited measurement data). For this thesis, the assumption that this system had an optimized state estimator was made. In that case, eigenstructure assignment technique can be fully implemented.

Table 13. Assumed Input Noise Intensity

	Noise direction (Input #)								
	1	2	3	4	5	6	7	8	9
Intensity	0	131.0617	0	0	0	0	19.9437	322.9485	0

Table 14. Optical Performance (Reduced Model Open-loop System)

Segment	D.O.F.	Open-Loop Jitter (Reduced Model)	Open-Loop Jitter (AFRL Model)	Units (rms)
A	piston	1773.2	46.68	nm
	X-tilt	143.7	168.44	nrad
	Y-tilt	144.8	143.03	nrad
B	piston	1136.6	45.64	nm
	X-tilt	105.3	106.57	nrad
	Y-tilt	112.8	89.96	nrad
C	piston	4368.3	60.58	nm
	X-tilt	172.4	170.5	nrad
	Y-tilt	122.1	117.88	nrad

Using the full-state feedback system, the target eigenvalue to be changed was determined to be the eigenvalue which is located in the frequency between 262 and 447 (rad / sec) as seen in the Figure 20.

After assigning the desired eigenvalue and using the open-loop system eigenvector as the desired eigenvector, the closed-loop system was developed and the resulting objective function value was 0.287×10^{-11} (12 % of open-loop system result) with max allowable gain value 1. The resulting singular value plot and closed-loop jitter performance was shown in Figure 21 and table 15.

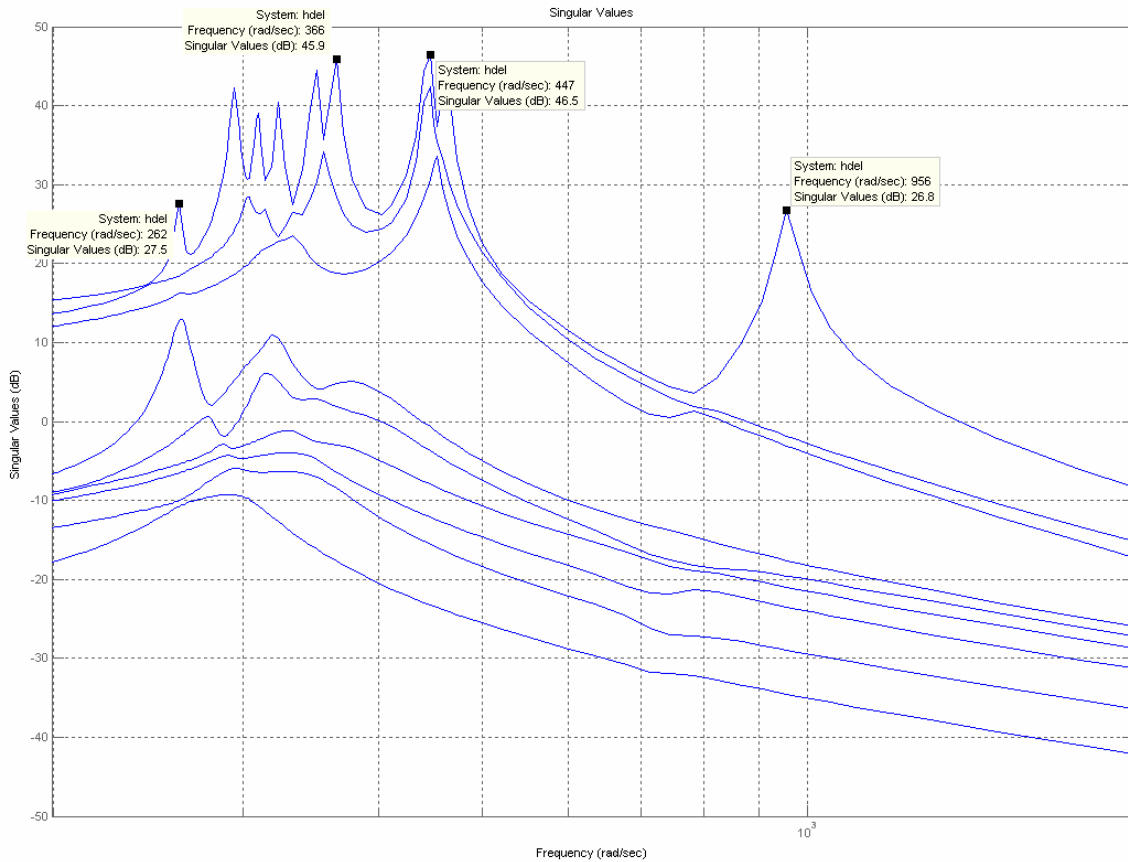


Figure 20. Open-Loop System Singular Plot (Reduced Model)

The proper desired eigenvector to be assigned was determined by using the first necessary condition which was described in Chapter 3, and given below:

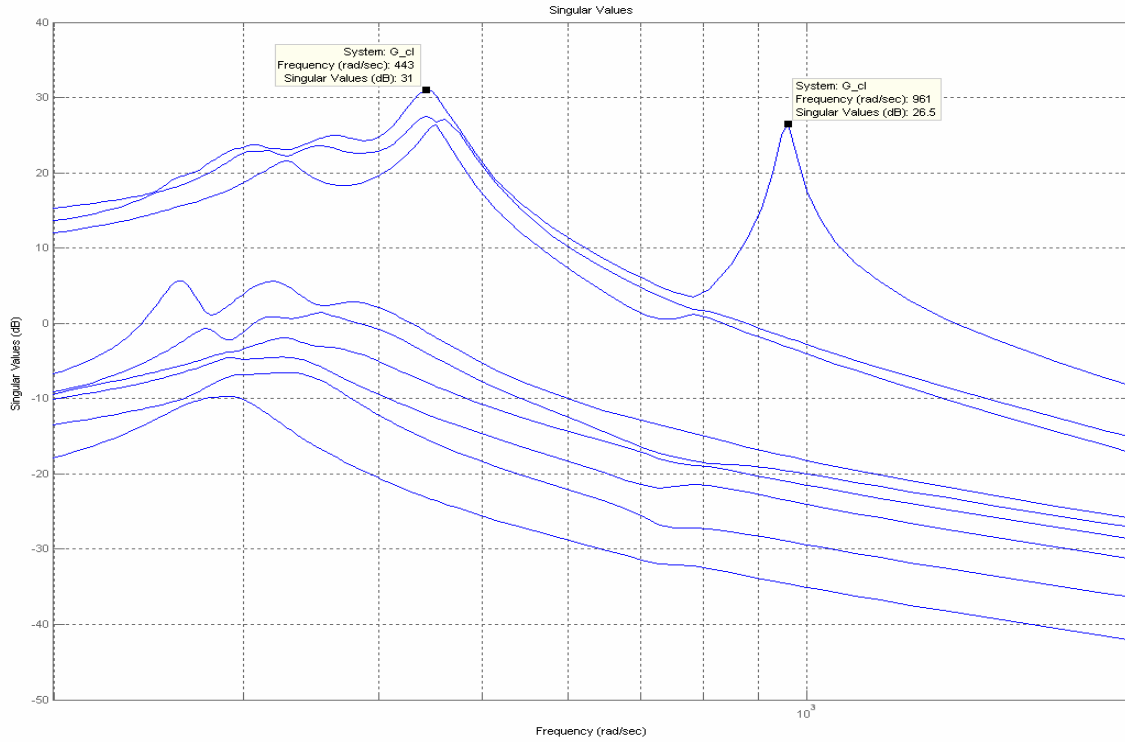
$$Vi_1 = [I_i I - A]^{-1} B \quad (103)$$

The second necessary condition wasn't used because this portion assumed full-state feedback.

After finding the optimum desired eigenvectors with related desired eigenvalue, the objective function result which satisfies the max allowable control gain matrix element value was almost the same with the result of assigning the desired eigenvalue.

Table 15. Optical Performance
(Reduced Model Closed-loop System with Assigning Eigenvalue)

Segment	D.O.F.	Closed-Loop Jitter (Reduced Model)	Open-Loop Jitter (Reduced Model)	Units (rms)
A	piston	598.4	1773.2	nm
	X-tilt	51.6	143.7	nrad
	X-tilt	52.2	144.8	nrad
B	piston	276	1136.6	nm
	X-tilt	22.5	105.3	nrad
	X-tilt	33.1	112.8	nrad
C	piston	1557.6	4368.3	nm
	X-tilt	53.9	172.4	nrad
	X-tilt	51.1	122.1	nrad



**Figure 21. Closed-Loop System Singular Plot
(Reduced Model after Assigning Eigenvalue)**

Summary

In analyzing the simplified 2-D model, the result between using eigenstructure assignment technique and using LQR method was almost the same. Both results showed 94.8% reduction of objective function. In analyzing the AFRL model, the objective function was reduced to 12% of the open-loop result.

Assigning the desired eigenvectors (i.e. shape control) took a great amount of control usage, but can reduce the effect of noise further. As described in assumptions, the result is highly dependent on the value of max allowable gain value. So as the control constraints change, the performance can be changed.

V. Conclusions and Recommendations

Conclusions

This research effort examined the use of eigenstructure assignment to maintain the focus for a sparse flexible array representative of a future space imaging system. It was expected that by increasing the real part of the eigenvalue of the target modes and controlling the eigenvector (i.e. shape control) to maintain the focus, eigenstructure assignment technique would lead to closed-loop control which minimized the effect of disturbance. The solution technique involved identifying a desired eigenstructure, and discussed methods of dealing with uncontrollable, unobservable modes, and the output feedback (rather than full-state feedback).

The advantage of the method was reduction of the computation time for solution. There were 2 parameters needed to be found in assigning the desired eigenvalue which was the most effective part in minimizing the effect of disturbance rather than 8 parameters in the LQR method. In finding the desired eigenvector, a search method was still needed but could reduce the computation time by using a basis for the achievable eigenvector range space which satisfied the necessary conditions.

In controlling the AFRL DOT model, which had less output measurements than states, the existence of a state estimator was essential to achieve the desired eigenvalues. If there is an optimum estimator, the eigenstructure assignment technique was a computationally faster approach than LQR.

Recommendations

In analyzing the AFRL model, this thesis assumed full-state feedback (i.e. there is optimized and perfect state estimators). This is impossible in a real system. Future

research on eigenstructure assignment techniques should take into account the existence of an estimator, and determine the effect on performance.

Not addressed in this thesis was the selection of a desired eigenvector (shape) to meet a secondary objective such as decoupling the ‘tip’ and ‘tilt’ modes. This approach could be used combined with optical modes (zernikes) to further enhance control by decoupling the optical modes.

For further studies, the planar model should be expanded to a 3-D model. Additionally flexible body effects of the individual mirrors could also be included to qualify these effects.

Appendix A: Main script file (Eigenstructure Assignment Technique)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Eigenstructure Assignment Technique
%   Planar model control by changing 3rd and 4th mode
%   Main script file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%driving the equations of motion for 2 mirrors and a base
clc;clear;close all;

I1= 1.5;
I2= 1.5;
Ib= 5;
m1= 1;
m2= 1;
mb= 50;
h= 4;
hm= 0.1;
L= 1;
l= 0.1;
k= 2.5*10^6;
c= 100;

%Initial Set up
phi=atan(h/L);
al_0=pi/4-phi/2 ;

DM=[1 0 -1 -l*cos(al_0) 0 (L+l*cos(al_0));
    1 0 -1 l*cos(al_0) 0 (L-l*cos(al_0));
    0 1 -1 0 -l*cos(al_0) -(L-l*cos(al_0));
    0 1 -1 0 l*cos(al_0) -(L+l*cos(al_0))];%distance matrix

DDM=[DM zeros(4,6);
     zeros(4,6) DM];

ACM=[k 0 0 0 c 0 0 0;
     0 k 0 0 0 c 0 0;
     0 0 k 0 0 0 c 0;
     0 0 0 k 0 0 0 c];

AB=[1/m1*[-1 -1 0 0];
    1/m2*[0 0 -1 -1];
    1/mb*[1 1 1 1];
    1/I1*l*cos(al_0)*[1 -1 0 0];

```

```

1/I2*I*cos(al_0)*[0 0 1 -1];
1/Ib*[(L+I*cos(al_0))*[-1 0 0 1]+(L-I*cos(al_0))*[0 -1 1 0]];

A=[zeros(6) eye(6)
  AB*ACM*DDM];
B=[zeros(6,4);
  AB];

Cm=[[cos(phi)/sqrt(L^2+h^2), 0, -cos(phi)/sqrt(L^2+h^2), 1, 0, -
sin(phi)*(h+hm)/sqrt(L^2+h^2) ; %mirror 1 tilt angle
0, -cos(phi)/sqrt(L^2+h^2), cos(phi)/sqrt(L^2+h^2), 0, 1, -
sin(phi)*(h+hm)/sqrt(L^2+h^2) ]; %mirror 2 tilt angle
-sin(phi) 0 sin(phi) tan(phi)*sqrt(L^2+h^2) 0 -
(h+hm)*cos(phi) ; %mirror 1 wavelength error
0 -sin(phi) sin(phi) 0 -tan(phi)*sqrt(L^2+h^2)
(h+hm)*cos(phi)]; %mirror 2 wavelength error

C=[Cm zeros(4,6);
  zeros(4,6) Cm];
D=zeros(8,4);

[vo do]=eig(A);do=diag(do);

%Controllable system Identification
[ABAR,BBAR,CBAR,T,KK]=ctrbf(A,B,C);
Ac=ABAR(5:12,5:12);
Bc=BBAR(5:12,:);
Cc=CBAR(:,5:12);
Gol=ss(Ac,Bc,Cc(1:4,:),D(1:4,:));

[vd,lamd]=eig(Ac);lamd=diag(lamd);
wn5=abs(lamd(5)); %Natural Frequency (Mode 4)
wn7=abs(lamd(7)); %Natural Frequency (Mode 3)

max_k=100
x0=[real(lamd(5));real(lamd(7))];

%Using Newton's Line Search Method
option=optimset('TolX',1e-12,'TolFun',1e-20,'TolCon',1e-
18,'MaxFunEvals',10^6,'MaxIter',10^4,'Display','iter');
[x,fval1,exitflag,output]=
fmincon(@fun_nock,x0,[],[],[],[],[],@nonlcon_nock,option,Ac,Bc,Cc,max_k);

%Overlapping Desired Eigenvalue
lamd(5)=x(1)+sqrt(wn5^2-x(1)^2)*i;

```

```

lamd(6)=x(1)-sqrt(wn5^2-x(1)^2)*i;
lamd(7)=x(2)+sqrt(wn7^2-x(2)^2)*i;
lamd(8)=x(2)-sqrt(wn7^2-x(2)^2)*i;

%Eigenstructure Assignment Technique
for jj=1:8
    N=[lamd(jj)*eye(8)-Ac, -Bc ,zeros(8,8);
        zeros(4,12)      ,Bc';
        eye(8)      ,zeros(8,4), (lamd(jj)*eye(8)-Ac)'];
    VWN(:,jj)=inv(N)*[zeros(12,1);vd(:,jj)];
end
VWN;

for jj=1:2:8
    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end
VWN;
% required gain
Kbar=-VWN(9:12,:)*pinv(Cc*VWN(1:8,:))

Sw=diag([1 1 1 1])*2*10^-7; % Assumed White Gaussian noise Intensity

%open loop root mean square(lqr)
S_x=lyap(Ac,Bc*Sw*Bc');% Variance of x matrix
S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
RMS_cl=diag(sqrt(S_y)); %Close loop Root mean square of x

p_angle=10^4;
P=diag([p_angle,p_angle,1,1]); %Penalty matrix
f_ol=RMS_cl'*P*RMS_cl*10^10

%closed loop root mean square(lqr)
S_x=lyap(Ac-Bc*Kbar*Cc,Bc*Sw*Bc');% Variance of x matrix
S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
RMS_cl=diag(sqrt(S_y)); %Close loop Root mean square of x

p_angle=10^4;
P=diag([p_angle,p_angle,1,1]); %Penalty matrix
f_cl=RMS_cl'*P*RMS_cl*10^10

%finding Optimum Desired Eigenvector

```



```

for ii=1:2:3
    lamd(ii+4)
    SUB=orth(inv(A-lamd(ii+4)*eye(12))*B);
    rank_sub=rank(SUB);
    OL=real(vo(7:12,[1,3,5,7]));
    OL_b=lamd(ii+4)*OL;
    SUB_2=[OL;OL_b];
    Rank_sub2=rank(SUB_2);
    SUB_total=[SUB SUB_2];
    rank_total=rank(SUB_total);

    x0=[1;1;1];
    option=optimset('TolX',1e-7,'TolFun',1e-7,'TolCon',1e-9,
        'MaxFunEvals',10^6,'MaxIter',10^4,'Display','iter');
    [x,fval,exitflag,output] =
    fmincon(@fun_vec_t,x0,[],[],[],[],[],@nonlcon_vec,option,SUB_2,C);
    x_f=[x;1];
    vd_opt(:,ii)=SUB_2*x_f;
end
    for kk=1:2:3
        vd_opt(:,kk)=vd_opt(:,kk)/norm(vd_opt(:,kk)); %Optimum Eigenvector
    end
    Trans=T*vd_opt;

% Assigning Proper Ratio between Open-Loop Eigenvector and Optimum Eigenvector

x0=[0.5;0.5];
option=optimset('TolX',1e-7,'TolFun',1e-7,'TolCon',1e-
9,'MaxFunEvals',10^6,'MaxIter',10^4,'Display','iter');
[x,fval,exitflag,output] =
fmincon(@fun_vec_a,x0,[],[],[],[],[],@nonlcon_vec_a,option,Ac,Bc,Cc,max_k,vd,lam
d,Trans);
vd(:,5)=x(1)*vd(:,5)+(1-x(1))*Trans(5:12,1);
vd(:,6)=conj(vd(:,5));
vd(:,7)=x(2)*vd(:,7)+(1-x(2))*Trans(5:12,3);
vd(:,8)=conj(vd(:,7));

for jj=1:8
    N=[lamd(jj)*eye(8)-Ac, -Bc ,zeros(8,8);
        zeros(4,12) ,Bc';
        eye(8) ,zeros(8,4), (lamd(jj)*eye(8)-Ac)'];
    VWN(:,jj)=inv(N)*[zeros(12,1);vd(:,jj)];
end

for jj=1:2:8

```

```

    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end

% required gain matrix
Kbar=-VWN(9:12,:)*pinv(Cc*VWN(1:8,:));
[vc dc]=eig(A-B*Kbar*C);dc=diag(dc);

%closed loop root mean square(lqr)
S_x=lyap(Ac-Bc*Kbar*Cc,Bc*Sw*Bc');% Variance of x matrix
S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
RMS_cl=diag(sqrt(S_y)); %Close loop Root mean square of x

f_cl=RMS_cl'*P*RMS_cl*10^10

```

Appendix B: Objective Function File (Finding Eigenvalue)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Eigenstructure Assignment Technique
%   Planar model control by 3rd and 4th mode change
%   Objective Function File
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```
function f=fun_nock(x,Ac,Bc,Cc,max_k)
```

```

[vd,lamd]=eig(Ac);lamd=diag(lamd);
wn5=abs(lamd(5)); %Natural Frequency (Mode 4)
wn7=abs(lamd(7)); %Natural Frequency (Mode 3)

```

```

lamd(5)=x(1)+sqrt(wn5^2-x(1)^2)*i;
lamd(6)=x(1)-sqrt(wn5^2-x(1)^2)*i;
lamd(7)=x(2)+sqrt(wn7^2-x(2)^2)*i;
lamd(8)=x(2)-sqrt(wn7^2-x(2)^2)*i;

```

```

for jj=1:8
    N=[lamd(jj)*eye(8)-Ac, -Bc,zeros(8,8);
      zeros(4,12),Bc';
      eye(8),zeros(8,4),(lamd(jj)*eye(8)-Ac)'];
    VWN(:,jj)=inv(N)*[zeros(12,1);vd(:,jj)];
end
VWN;

```

```

for jj=1:2:8
    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end
VWN;
% required gain
Kbar=-VWN(9:12,:)*pinv(Cc*VWN(1:8,:));

```

```

E=eig(Ac-Bc*Kbar*Cc);E=real(E);I=find(E>0);s=size(I);
CON=cond(Ac-Bc*Kbar*Cc);

```

```

if CON<10^9 & s(1)==0
    Sw=diag([1 1 1 1])*2*10^-7; % Assumed White Gaussian noise Intensity

    %closed loop root mean square(lqr)
    S_x=lyap(Ac-Bc*Kbar*Cc,Bc*Sw*Bc'); % Variance of x matrix
    S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
    RMS_cl=diag(sqrt(S_y)); %Close loop Root mean square of x

    P=diag([10^4,10^4,1,1]); %Penalty matrix
    f=RMS_cl'*P*RMS_cl*10^10; %Objective Function
else
    f=10^20;
end

```

Appendix C: Constraint Function File (Finding Eigenvalue)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Eigenstructure Assignment Technique
%   Planar model control by 3rd and 4th mode change
%   Constraint Function File
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [c,ceq]=nonlcon_nock(x,Ac,Bc,Cc,max_k)

[vd,lamd]=eig(Ac);lamd=diag(lamd);
wn5=abs(lamd(5)); %Natural Frequency (Mode 4)
wn7=abs(lamd(7)); %Natural Frequency (Mode 3)

lamd(5)=x(1)+sqrt(wn5^2-x(1)^2)*i;
lamd(6)=x(1)-sqrt(wn5^2-x(1)^2)*i;
lamd(7)=x(2)+sqrt(wn7^2-x(2)^2)*i;
lamd(8)=x(2)-sqrt(wn7^2-x(2)^2)*i;

for jj=1:8
    N=[lamd(jj)*eye(8)-Ac, -Bc ,zeros(8,8);
        zeros(4,12)      ,Bc';
        eye(8)      ,zeros(8,4), (lamd(jj)*eye(8)-Ac)'];
    VWN(:,jj)=inv(N)*[zeros(12,1);vd(:,jj)];
end

for jj=1:2:8
    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end

% required gain
Kbar=-VWN(9:12,:)*pinv(Cc*VWN(1:8,:));

for jj=1:8
    temp(4*(jj-1)+1:4*(jj-1)+4,1)=Kbar(:,jj);
end
temp=abs(temp);
c=[temp-max_k*ones(32,1);
    x];
ceq=[];

```

Appendix D: Objective Function File (Finding Eigenvector)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Planar model control By Eigenstructure Assignment
%   Objective Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function f=fun_vec_a(x,Ac,Bc,Cc,max_k,vd,lamd,Trans)

vd(:,5)=x(1)*vd(:,5)+(1-x(1))*Trans(5:12,1);
vd(:,6)=conj(vd(:,5));
vd(:,7)=x(2)*vd(:,7)+(1-x(2))*Trans(5:12,3);
vd(:,8)=conj(vd(:,7));

for jj=1:8
    N=[lamd(jj)*eye(8)-Ac, -Bc ,zeros(8,8);
        zeros(4,12)      ,Bc';
        eye(8)      ,zeros(8,4), (lamd(jj)*eye(8)-Ac)'];
    VWN(:,jj)=inv(N)*[zeros(12,1);vd(:,jj)];
end
VWN;

for jj=1:2:8
    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end
VWN;
% required gain
Kbar=-VWN(9:12,:)*pinv(Cc*VWN(1:8,:));

%closed loop root mean square(lqr)
Sw=diag([1 1 1 1])*2*10^-7; % Assumed White Gaussian noise Intensity
S_x=lyap(Ac-Bc*Kbar*Cc,Bc*Sw*Bc');% Variance of x matrix
S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
RMS_cl=diag(sqrt(S_y)); %Close loop Root mean square of x
p_angle=10^4;
P=diag([p_angle,p_angle,1,1]); %Penalty matrix
f=RMS_cl'*P*RMS_cl*10^10 ;

```

Appendix E: Constraint Function File (Finding Eigenvector)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Planar model control By Eigenstructure Assignment
%   Constraint Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [c,ceq]=nonlcon_vec_a(x,Ac,Bc,Cc,max_k,vd,lamd,Trans)

vd(:,5)=x(1)*vd(:,5)+(1-x(1))*Trans(5:12,1);
vd(:,6)=conj(vd(:,5));
vd(:,7)=x(2)*vd(:,7)+(1-x(2))*Trans(5:12,3);
vd(:,8)=conj(vd(:,7));

for jj=1:8
    N=[lamd(jj)*eye(8)-Ac, -Bc ,zeros(8,8);
        zeros(4,12)      ,Bc';
        eye(8)      ,zeros(8,4), (lamd(jj)*eye(8)-Ac)'];
    VWN(:,jj)=inv(N)*[zeros(12,1);vd(:,jj)];
end

for jj=1:2:8
    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end

% required gain
Kbar=-VWN(9:12,:)*pinv(Cc*VWN(1:8,:));

m_k=max(max(abs(Kbar)));

c=[m_k-max_k;
    -x;
    x-ones(2,1)];

ceq=[];

```

Appendix F: Main script file (LQR)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%
%   Planar model control By L Q R Method
%   Main script file
%   %%%%%%%%%%
%%
%driving the equations of motion for 2 mirrors and a base
clc
clear
close all

I1= 1.5;
I2= 1.5;
Ib= 5;
m1= 1;
m2= 1;
mb= 50;
h= 4;
hm= 0.1;
L= 1;
l= 0.1;
k= 2.5*10^6;
c= 100;

%Initial Set up
phi=atan(h/L);
al_0=pi/4-phi/2 ;

DM=[1 0 -1 -l*cos(al_0) 0 (L+l*cos(al_0));
    1 0 -1 l*cos(al_0) 0 (L-l*cos(al_0));
    0 1 -1 0 -l*cos(al_0) -(L-l*cos(al_0));
    0 1 -1 0 l*cos(al_0) -(L+l*cos(al_0))];%distance matrix

DDM=[DM zeros(4,6);
     zeros(4,6) DM];

ACM=[k 0 0 0 c 0 0 0;
     0 k 0 0 0 c 0 0;
     0 0 k 0 0 0 c 0;
     0 0 0 k 0 0 0 c];

```



```

AB=[1/m1*[-1 -1 0 0];
    1/m2*[0 0 -1 -1];
    1/mb*[1 1 1 1];
    1/l1*l*cos(al_0)*[1 -1 0 0];
    1/l2*l*cos(al_0)*[0 0 1 -1];
    1/lb*[(L+l*cos(al_0))*[-1 0 0 1]+(L-l*cos(al_0))*[0 -1 1 0]]];

A=[zeros(6) eye(6)
    AB*ACM*DDM];
B=[zeros(6,4);
    AB];

Cm=[[cos(phi)/sqrt(L^2+h^2), 0, -cos(phi)/sqrt(L^2+h^2), 1, 0, -
sin(phi)*(h+hm)/sqrt(L^2+h^2) ; %mirror 1 tilt angle
    0, -cos(phi)/sqrt(L^2+h^2), cos(phi)/sqrt(L^2+h^2), 0, 1, -
sin(phi)*(h+hm)/sqrt(L^2+h^2) ]; %mirror 2 tilt angle
    [-sin(phi) 0 sin(phi) tan(phi)*sqrt(L^2+h^2) 0 -
(h+hm)*cos(phi) ; %mirror 1 wavelength error
    0 -sin(phi) sin(phi) 0 -tan(phi)*sqrt(L^2+h^2)
(h+hm)*cos(phi)]]; %mirror 2 wavelength error

C=[Cm zeros(4,6);
    zeros(4,6) Cm];
D=zeros(8,4);

[vo do]=eig(A);do=diag(do);

%Controllable system Identification
[ABAR,BBAR,CBAR,T,KK] =ctrbf(A,B,C);
Ac=ABAR(5:12,5:12);
Bc=BBAR(5:12,:);
Cc=CBAR(:,5:12);

max_k=100
x=[100;100;100;100;1;1;.01;.01]

option=optimset('TolX',1e-10,'TolFun',1e-19,'TolCon',1e-
10,'MaxFunEvals',10^6,'MaxIter',10^5,'Display','iter');

for kkk=1:1
load XXX0
x0=(x*1)/1;
[x,fval1,exitflag,output] =
fmincon(@fun_lqr,x0,[],[],[],[],[],@nonlcon_lqr,option,Ac,Bc,Cc,max_k);
%save XXX0 x;

```

end

```
Q=diag([x])*10^1;  
R=diag([1,1,1,1]*.1);  
K=lqr(Ac,Bc,Q,R);
```

```
Kbar=K*Cc^-1;
```

```
Sw=diag([1 1 1 1])*2*10^-7; % Assumed White Gaussian noise Intensity
```

```
%closed loop root mean square(lqr)
```

```
S_x=lyap(Ac-Bc*Kbar*Cc,Bc*Sw*Bc'); % Variance of x matrix
```

```
S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
```

```
RMS_cl=diag(sqrt(S_y)); % Close loop Root mean square of x
```

```
P=diag([10^4,10^4,1,1]); %Penalty matrix
```

```
f=RMS_cl'*P*RMS_cl*10^10
```

Appendix G: Objective Function File (LQR)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Planar model control By L Q R Method
%   Objective Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```
function f=fun_lqr(x,Ac,Bc,Cc,max_k)
```

```

Q=diag([x])*10^1; % State Penalty
R=diag([1,1,1,1]*.1);% Control Penlaty
K=lqr(Ac,Bc,Q,R);

```

```
Kbar=K*Cc^-1;
```

```
Sw=diag([1 1 1 1])*2*10^-7; % Assumed White Gaussian noise Intensity
```

```

%closed loop root mean square(lqr)
S_x=lyap(Ac-Bc*Kbar*Cc,Bc*Sw*Bc');% Variance of x matrix
S_y=Cc(1:4,:)*S_x*Cc(1:4,:); % Variance of y matrix
RMS_cl=diag(sqrt(S_y)); %Close loop Root mean square of x

```

```

P=diag([10^4,10^4,1,1]); %Penalty matrix
f=RMS_cl'*P*RMS_cl*10^10;

```

Appendix H: Constraint Function File (LQR)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Planar model control By L Q R Method
%   Constraint Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [c,ceq]=nonlcon_lqr(x,Ac,Bc,Cc,max_k)

Q=diag([x])*10^1; % State Penalty
R=diag([1,1,1,1]*.1);% Control Penlaty
K=lqr(Ac,Bc,Q,R);

Kbar=K*Cc^-1;

for jj=1:8
    temp(4*(jj-1)+1:4*(jj-1)+4,1)=Kbar(:,jj);
end
temp=abs(temp);
c=[temp-max_k*ones(32,1);
    -x+0.1*ones(8,1)];

ceq=[];
```

Appendix I: Main script file (AFRL Model)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   AFRL Model control By Eigenstructure Assignment Technique
%   Main script file
%   %%%%%%%%%%%%%
%%
clc;clear;close all;

% Model Reduction
load tuned_best_fit_0L
sys=d2c(fit_sys);
n=size(sys(:,1));
h=minreal(sys);

[hb,g]=balreal(h);

index=21:79;
hdel=modred(hb,index,'del');
figure(1);sigma(hdel,{200,2000});grid on
[A,B,C,D]=ssdata(hdel);

%Setting Open Loop Value as desired one.
[vd lamd]=eig(A);lamd=diag(lamd);
[xx0,I]=sort(abs(lamd));lamdd=lamd;vdd=vd;

%Arranging Eigenvalue by increasing order
for kk=1:20
    lamd(kk)=lamdd(I(kk));
    vd(:,kk)=vdd(:,I(kk));
end
for kk=1:9
    r(kk,1)=real(lamd(2*kk));
    Wn(kk,1)=abs(lamd(2*kk));
end

max_k=1                %Max allowable gain matrix element.

%Open loop root mean square(lqr)
load XX_modred          % Contains variable for the White Gaussian noise
intensity (x)

```

```

Sw=diag(x); % White Gaussian noise intensity

S_x=lyap(A,B*Sw*B'); % Variance of x matrix
S_y=C*S_x*C'; % Variance of y matrix
RMS1=diag(sqrt(S_y)) % Closed loop Root mean square of x
f=RMS1'*RMS1/10^4 % Objective Function

%Finding optimum eigenvalue assignment with open loop eigenvector for desired one.
x0=[r]
option=optimset('TolX',1e-9,'TolFun',1e-9,'TolCon',1e-9,
'MaxFunEvals',10^6,'MaxIter',40^3,'Display','iter');
[x,fval1,exitflag,output] =
fmincon(@fun_e_val,x0,[],[],[],[],[],[],@nonlcon_e_val,option,A,B,C,max_k,lamd,vd,
Wn);
save result_e_val x

for kk=1:9
    lamd(2*kk-1)=x(kk,1)+sqrt(Wn(kk)^2-x(kk,1)^2)*i;
    lamd(2*kk)=conj(lamd(2*kk-1));
end

% Checking the result of assigning the eigenvalue.
P=eye(20); %eigenvector penalty matrix

for jj=1:20
    N=[lamd(jj)*eye(20)-A, -B ,zeros(20);
        zeros(9,20+9) ,B';
        P ,zeros(20,9), (lamd(jj)*eye(20)-A)'];
    VWN(:,jj)=inv(N)*[zeros(29,1);P*vd(:,jj)];
end

for jj=1:2:20
    VWN(:,jj)=real(VWN(:,jj));
    VWN(:,jj+1)=imag(VWN(:,jj+1));
end

Kbar=-VWN(20+1:20+9,:)*pinv(eye(20)*VWN(1:20,:)); % required gain matrix with
full state-feedback
G_cl=ss(A-B*Kbar,B,C,D);
figure(2);sigma(G_cl,{200,2000});grid on

%Closed loop root mean square(lqr)
load XX_modred % Contains variable for the White Gaussian noise
intensity (x)

```

```

Sw=diag(x); % White Gaussian noise intensity
S_x=lyap(A-B*Kbar,B*Sw*B'); % Variance of x matrix
S_y=C*S_x*C'; % Variance of y matrix
RMS=diag(sqrt(S_y)) % Closed loop Root mean square of x
f=RMS'*RMS/10^11
MAX_K=max(max(abs(Kbar))) % Objective Function
save E_VAL_result

```

Appendix J: Objective Function File(AFRL Model)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   AFRL Model control By Eigenstructure Assignment Technique
%   Objective Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function f=fun_e_val(x,A,B,C,max_k,lamd,vd,Wn)

    for kk=1:9
        lamd(2*kk-1)=x(kk,1)+sqrt(Wn(kk)^2-x(kk,1)^2)*i;
        lamd(2*kk)=conj(lamd(2*kk-1));
    end

    % Checking the result of assigning the eigenvalue.
    P=eye(20); %eigenvector penalty matrix

    for jj=1:20
        N=[lamd(jj)*eye(20)-A, -B ,zeros(20);
          zeros(9,20+9)      ,B';
          P      ,zeros(20,9), (lamd(jj)*eye(20)-A)'];
        VWN(:,jj)=inv(N)*[zeros(29,1);P*vd(:,jj)];
    end

    for jj=1:2:20
        VWN(:,jj)=real(VWN(:,jj));
        VWN(:,jj+1)=imag(VWN(:,jj+1));
    end

    Kbar=-VWN(20+1:20+9,:)*pinv(eye(20)*VWN(1:20,:)); % required gain

    E=eig(A-B*Kbar);E=real(E);I=find(E>0);s=size(I);
    CON=cond(A-B*Kbar);
    if CON<10^9 & s(1)==0

        load XX_modred % Contains variable for the White Gaussian noise intensity
        Sw=diag(x); % White Gaussian noise intensity
        %Closed loop root mean square(lqr)
        S_x=lyap(A-B*Kbar,B*Sw*B'); % Variance of x matrix
        S_y=C*S_x*C'; % Variance of y matrix
        RMS=diag(sqrt(S_y)); %Closed loop Root mean square of x
        f=RMS'*RMS/10^4;
    end

```



```
else  
    f=10^20;  
end
```

Appendix K: Constraint Function File (AFRL Model)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   AFRL Model control By Eigenstructure Assignment Technique
%   Constraint Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [c,ceq]=nonlcon_e_val(x,A,B,C,max_k,lamd,vd,Wn)

    for kk=1:9
        lamd(2*kk-1)=x(kk,1)+sqrt(Wn(kk)^2-x(kk,1)^2)*i;
        lamd(2*kk)=conj(lamd(2*kk-1));
    end

    % Checking the result of assigning the eigenvalue.
    P=eye(20); %eigenvector penalty matrix

    for jj=1:20
        N=[lamd(jj)*eye(20)-A, -B ,zeros(20);
          zeros(9,20+9)      ,B';
          P      ,zeros(20,9), (lamd(jj)*eye(20)-A)'];
        VWN(:,jj)=inv(N)*[zeros(29,1);P*vd(:,jj)];
    end

    for jj=1:2:20
        VWN(:,jj)=real(VWN(:,jj));
        VWN(:,jj+1)=imag(VWN(:,jj+1));
    end

    Kbar=-VWN(20+1:20+9,:)*pinv(eye(20)*VWN(1:20,:)); % required gain matrix

    K_max=max(max(abs(Kbar)));

    c=[x(:,1);K_max-max_k];
    ceq=[];

```

Appendix L: Objective Function File (AFRL Model)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   AFRL Model control By Eigenstructure Assignment Technique
%   Objective Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function f=fun_e_val(x,A,B,C,max_k,lamd,vd,Wn)

    for kk=1:9
        lamd(2*kk-1)=x(kk,1)+sqrt(Wn(kk)^2-x(kk,1)^2)*i;
        lamd(2*kk)=conj(lamd(2*kk-1));
    end

    % Checking the result of assigning the eigenvalue.
    P=eye(20); %eigenvector penalty matrix

    for jj=1:20
        N=[lamd(jj)*eye(20)-A, -B ,zeros(20);
          zeros(9,20+9)      ,B';
          P      ,zeros(20,9), (lamd(jj)*eye(20)-A)'];
        VWN(:,jj)=inv(N)*[zeros(29,1);P*vd(:,jj)];
    end

    for jj=1:2:20
        VWN(:,jj)=real(VWN(:,jj));
        VWN(:,jj+1)=imag(VWN(:,jj+1));
    end

    Kbar=-VWN(20+1:20+9,:)*pinv(eye(20)*VWN(1:20,:)); % required gain

    E=eig(A-B*Kbar);E=real(E);I=find(E>0);s=size(I);
    CON=cond(A-B*Kbar);
    if CON<10^9 & s(1)==0

        load XX_modred % Contains variable for the White Gaussian noise intensity
        Sw=diag(x); % White Gaussian noise intensity
        %Closed loop root mean square(lqr)
        S_x=lyap(A-B*Kbar,B*Sw*B'); % Variance of x matrix
        S_y=C*S_x*C'; % Variance of y matrix
        RMS=diag(sqrt(S_y)); %Closed loop Root mean square of x
        f=RMS'*RMS/10^4;
    end

```

```
else  
    f=10^20;  
end
```

Appendix M: Constraint Function File (AFRL Model)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   AFRL Model control By Eigenstructure Assignment Technique
%   Constraint Function file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [c,ceq]=nonlcon_e_val(x,A,B,C,max_k,lamd,vd,Wn)

    for kk=1:9
        lamd(2*kk-1)=x(kk,1)+sqrt(Wn(kk)^2-x(kk,1)^2)*i;
        lamd(2*kk)=conj(lamd(2*kk-1));
    end

    % Checking the result of assigning the eigenvalue.
    P=eye(20); %eigenvector penalty matrix

    for jj=1:20
        N=[lamd(jj)*eye(20)-A, -B ,zeros(20);
          zeros(9,20+9)      ,B';
          P      ,zeros(20,9), (lamd(jj)*eye(20)-A)'];
        VWN(:,jj)=inv(N)*[zeros(29,1);P*vd(:,jj)];
    end

    for jj=1:2:20
        VWN(:,jj)=real(VWN(:,jj));
        VWN(:,jj+1)=imag(VWN(:,jj+1));
    end

    Kbar=-VWN(20+1:20+9,:)*pinv(eye(20)*VWN(1:20,:)); % required gain matrix

    K_max=max(max(abs(Kbar)));

    c=[x(:,1);K_max-max_k];
    ceq=[];

```

Bibliography

- [1] Karl N. Schrader, Robert H. Fetner, Steven F. Griffin, Richard Scott Erwin Boeing-SVS, Inc. *Development of a sparse-aperture testbed for opto-mechanical control of space-deployable structures*, Air Force Research Laboratory.
- [2] Katsuhiko Ogata. *Modern Control Engineering Fourth Edition*, Prentice Hall, Inc, 2002.
- [3] Karl N. Schrader, Robert H. Fetner, Jed Donaldson, Robert J. Fuentes, Richard Scott Erwin .Boeing-SVS, Inc.; *Integrated control system development for phasing and vibration suppression for a sparse-array telescope*. Air Force Research Laboratory.
- [4] Huckabone, Thomas C. *Algorithm for Robust Eigenstructure Assignment Using the LQR*. MS thesis, Air Force Institute of Technology, Wright Patterson AFB,OH,December 1991
- [5] Meirovitch, Leonard. *Element of Vibration analysis*. New York: Mc Graw Hill Publishing Co, Inc,1986.
- [6] Robinson, J. D. *A Linear Quadratic Regulator Weight Selection Algorithm for Robust Pole Assignment*. MS thesis, Air Force Institute of Technology, Wright Patterson AFB,OH, 1990
- [7] Jeffrey B. Burl. *Linear Optimal Control*, Addison Wesley Longman, Inc,1999.
- [8] Arther E. Bryson, Jr. *Dynamic Optimization*, Addison Wesley Longman, Inc,1999.
- [9] David Poole. *Linear Algebra (A Modern Introduction)*, Thomson Learning, Inc,2003.
- [10] Harris, Adam G. *Robust Eigenstructure Assignment Using Positive Definite Output Feedback Control*. MS thesis, Air Force Institute of Technology, Wright Patterson AFB, OH, December 1999.
- [11] Richard. Cobb, *Lecture notes*. Air Force Institute of Technology, Wright Patterson AFB,OH
- [12] *Matlab Control System Toolbox User Manual*. Lyap. Limitation. Mathworks.2001.
- [13] M.K. Powers, E.K. Hall, R.S. Erwin, “*UltraLITE Deployable Optical Telescope (DOT) Technical Requirements Document – Release 5.7,*” Air Force Research Laboratory, AFRL-VS-TR-2001-1078, May 9, 2001.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 074-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) 09-03-2004		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From – To) Sep 2002 – Mar 2004	
4. TITLE AND SUBTITLE CONTROLLING THE PRIMARY MIRROR IN A SPACE-BASED TELESCOPE UTILIZING AN EIGENSTRUCTURE ASSIGNMENT TECHNIQUE				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Kim Kyongsoo, Captain, ROK Army				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way, Building 641 WPAFB OH 45433-7765				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GAE/ENY/04-M10	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/VSSV Dr. Scott Erwin 3550 Aberdeen Ave. SE Kirtland AFB, NM 87117-5776				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT <p>The feasibility of using a sparse array of mirrors in a satellite system is being evaluated. This study focuses on the control laws necessary for achieving the minimization of the effect of vibration which happens at the primary mirror of the satellite, by using the eigenstructure assignment technique. White Gaussian noise was assumed as the external input. AFRL has developed a 79 state model with 9 control inputs and 9 outputs, 3 sparse array mirrors, and a base. To extend the theory, a simplified model with 2 mirrors and a base which has 12 states, 4 control inputs and 8 outputs was developed. The system is not completely controllable nor observable because there are 4 rigid body modes.</p> <p>The control law starts from realizing the states which are controllable or uncontrollable. Then only for the controllable states, the eigenstructure assignment technique is applied. The closed-loop eigenstructure was determined by analyzing the open-loop modes. To find the best value for the eigenstructure, Newton's line search method was applied.</p> <p>The relative motions of the primary and secondary mirrors were described by wave reflection dynamics. Results show that this technique achieves a reasonable amount of control use for the objective and relates these result to those obtained from linear quadratic regulator (LQR) technique.</p>					
15. SUBJECT TERMS Eigenstructure Assignment Technique, LQR, Lyapunov Function, Riccati Equation,					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES 103	19a. NAME OF RESPONSIBLE PERSON Richard G. Cobb, Major USAF (ENY)
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (Include area code) 937-255-3069; e-mail: richard.cobb@afit.edu